# Wind pattern analysis applied to Tokyo 2020 Olympic Games


**Fabio Di Francesco**

Supervisor: Prof. Alicia Ageno

Department of Computer Science
Universitat Politécnica de Catalunya

This dissertation is submitted for the degree of
*Master in Innovation and Research in Informatics*

Universitat Politécnica de Catalunya

October 2018

# Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements.

Fabio Di Francesco
October 2018

# Acknowledgements

I would like to thank Professor Ageno for the great help in the work of this thesis, as well as TriM team, that provided knowledge and suggestions about this field that was new to me, and also Meteocat that was always available to give information and explanation on the data used that allowed to develop the implementation of this project.

# Abstract

The following master thesis is the product of the work carried out during the Erasmus exchange of the year 2017-2018 that involved the author, exchange student from the University of Bologna, the Universitat Politècnica de Catalunya , TriM, an italian company with a strong knowledge of weather data and forecasting, and Meteocat, the public meteorological company of Catolonia in a collaboration aimed to find new methodologies for the processing of meteorological data.

The reason that motivated this work is dictated by the increasing amount of weather data available today, that necessarily drives the weather forecasting in a more automated procedure that reduces the time needed to generate a forecast and the intervention of a human, in the figure of a meteorologist, in the analysis of the data. This allows to process more data and thus having predictions that take advantages of the usage of many information that could result in improved forecasting.

The development in the field of machine learning allows today to treat a vast amount of information in an automatic way, leaving the analysis process to the machines, freeing the user of this time consuming task. And unsupervised learning is the branch that can process data that are not labelled nor preprocessed, speeding up the data mining.

The goal of this thesis is to apply unsupervised learning techniques to this scope, taking inspiration from available literature that experimented in this field and combining different solutions into a new technique that aims to reduce the human decision in the process of the recognition of wind patterns and improve the automation of the whole process.

# Table of contents

**List of figures** **xiii**

**List of tables** **xv**

**Nomenclature** **xvii**

**1 Introduction** **1**
1.1 Meteorology . . . 1
1.2 The scope of the thesis . . . 1
1.3 The motivation . . . 2
1.4 Objectives . . . 2
1.5 Structure of the solution . . . 4
1.6 Next chapters . . . 5

**2 The meteorological data** **7**
2.1 Data collection . . . 7
2.2 Types of data . . . 7
2.3 Data sources . . . 9
2.3.1 Numerical Weather Prediction . . . 9
2.3.1.1 WRF . . . 9
2.3.1.2 AROME . . . 10
GRIB files . . . 11
netCDF . . . 12
2.3.2 Area of measurements . . . 18

**3 Clustering** **21**
3.1 Machine Learning . . . 21
3.2 Fundamentals of clustering . . . 22
3.3 Hierarchical Clustering . . . 24

3.3.1 The algorithm . . . 25
3.4 K-means . . . 27
3.4.1 K-means variant . . . 29
3.5 Distance measure . . . 31
3.6 Automatic and manual clustering . . . 31
3.6.1 Manual clustering . . . 32
3.6.2 Automatic clustering . . . 33
3.6.2.1 Quality measures . . . 34
3.7 Clustering comparison . . . 36
3.8 Classification . . . 38

**4 The implementation** **39**
4.1 First attempts . . . 39
4.2 Data Loading . . . 40
4.3 Normalisation . . . 42
4.4 Hierarchical Clustering . . . 43
4.4.1 Automatic clustering . . . 43
4.4.2 Manual clustering . . . 45
4.5 K-means . . . 47
4.6 Results report . . . 49
4.7 Classification . . . 51

**5 Results Analysis** **53**
5.1 Meteorological analysis of clustering results . . . 53
5.1.1 Automatic clustering . . . 55
5.1.2 Manual clustering . . . 57
5.1.3 Automatic clustering revisited . . . 58
5.2 Classification results . . . 60

**6 Conclusion** **65**
6.1 Future Work . . . 66

**References** **71**

**Appendix A Automatic clustering results** **75**
A.1 Automatic clustering infos . . . 75
A.2 K-means infos . . . 76
A.3 Clustering Results . . . 76

**Appendix B Manual clustering results** **91**
B.1 Manual clustering infos . . . . . 91
B.2 K-means infos . . . . . 91
B.3 Manual clustering results, k = 17 . . . . . 92
B.3.1 Threshold . . . . . 92
B.3.2 Comparison with automatic clustering . . . . . 107
B.3.3 Comparison with manual clustering k = 10 . . . . . 107
B.4 Manual clustering results, k = 10 . . . . . 109
B.4.1 Threshold . . . . . 109
B.4.2 Comparison with automatic clustering . . . . . 119
B.4.3 Comparison with manual clustering k = 17 . . . . . 119

# List of figures


1.1 Representation of the components of the solution . . . . . 4

2.1 Vector representation of the wind flow . . . . . 8
2.2 The area covered by AROME . . . . . 11
2.3 Screenshot from PanoplyJ . . . . . 15
2.4 Georeferenced Plot . . . . . 16
2.5 Representation of the values stored in the NetCDF file . . . . . 17
2.6 The 16 points chosen for WRF . . . . . 18
2.7 The 16 points chosen for AROME . . . . . 19

3.1 Examples of dendrograms . . . . . 25

4.1 Initial screen of the application. . . . . 40
4.2 Section of data normalisation. . . . . 42
4.3 Section of hierarchical clustering. . . . . 43
4.4 Section of automatic hierarchical clustering. . . . . 43
4.5 Section of manual hierarchical clustering. . . . . 46
4.6 Section of k-means. . . . . 47
4.7 Section of results report and analysis. . . . . 49

# List of tables


2.1 Example of WRF file . . . . . 10

3.1 Types of linkage . . . . . 26
3.2 Quality measures . . . . . 36

5.1 Clusters division by direction . . . . . 57
5.2 Clusters matching . . . . . 61
5.4 Clusters similarity . . . . . 62
5.4 Clusters similarity . . . . . 63

A.1 Clusters information . . . . . 77
A.2 Clusters information . . . . . 78
A.3 Clusters information . . . . . 79
A.4 Ranges of TEMPERATURE (in °C): . . . . . 80
A.5 Ranges of HUMIDITY (in %): . . . . . 81
A.6 Ranges of PRECIPITATION (in Kg/mˆ2): . . . . . 82
A.7 Ranges of PRESSURE (in Pa): . . . . . 83
A.8 Wind speed ranges (in m/s) . . . . . 84
A.9 Wind direction ranges . . . . . 85
A.10 Wind direction ranges . . . . . 86
A.11 Wind direction wider ranges . . . . . 87
A.12 Transition matrix . . . . . 88

B.1 Clusters information . . . . . 94
B.2 Clusters information . . . . . 95
B.3 Clusters information . . . . . 96
B.4 Ranges of TEMPERATURE (in °C): . . . . . 97
B.5 Ranges of HUMIDITY (in %): . . . . . 98
B.6 Ranges of PRECIPITATION (in Kg/mˆ2): . . . . . 99

B.7 Ranges of PRESSURE (in Pa): . . . . . 100
B.8 Wind speed ranges (in m/s) . . . . . 101
B.9 Wind direction ranges . . . . . 102
B.10 Wind direction wider ranges . . . . . 103
B.11 Wind direction wider ranges . . . . . 104
B.12 Transition matrix . . . . . 105
B.13 Clusters matching . . . . . 107
B.14 Clusters matching . . . . . 107
B.15 Clusters information . . . . . 111
B.16 Clusters information . . . . . 111
B.17 Clusters information . . . . . 112
B.18 Ranges of TEMPERATURE (in °C): . . . . . 112
B.19 Ranges of HUMIDITY (in %): . . . . . 113
B.20 Ranges of PRECIPITATION (in Kg/mˆ2): . . . . . 113
B.21 Ranges of PRESSURE (in Pa): . . . . . 114
B.22 Wind speed ranges (in m/s) . . . . . 114
B.23 Wind direction ranges . . . . . 115
B.24 Wind direction ranges . . . . . 116
B.25 Wind direction wider ranges . . . . . 116
B.26 Transition matrix . . . . . 117
B.27 Clusters matching . . . . . 119
B.28 Clusters matching . . . . . 119

# Nomenclature

**Acronyms / Abbreviations**

AROME Applications of Research to Operations at MEsoscale

NWP Numerical Weather Prediction

WRF Weather Research and Forecasting

# Chapter 1

# Introduction

## 1.1 Meteorology

Meteorology is the science that studies the atmosphere and the events connected to it. One of the main focus of meteorology is to produce weather forecasting, that is the application of the principles of meteorology in order to predict the phenomena that will happen in a given place and time. Various data are collected from instruments and sensors, and their changes over the time are studied to understand the meteorological phenomenon that is taking place and to construct models to be used for the prediction. This process requires the interpretation of a human, to choose the right model for the prediction and to interpret the results.

Weather forecasting is not only delivered to the general public, but there are some specific sectors that needs forecast to operate and grant safety. For example air traffic needs accurate weather forecasting to plan airplane routes in order to avoid thunderstorm or prevent icing of the wings. Furthermore forecasting is helpful to prevent and control wildfires.

Lastly an important application of weather forecasting is for navigation in waterways as weather can strongly influence the safety of the transit due to the wind, waves and tides. So in this case the wind plays a fundamental role, along with other weather parameters.

## 1.2 The scope of the thesis

The scope of the thesis focuses on marine weather forecasting, particularly in the study of wind patterns. The present master thesis is developed in the framework of the

Tokyo2020 Olympic Games Weather Project, leaded by TriM company and funded by the Austrian Sailing Federation and by Croatia and Cyprus Laser Olympic classes. Sailing strategy and performance are strongly related with environmental parameters such as weather, oceanic current and geographical data. A thorough prediction of the conditions expected during a sailing race is a valuable information for a sailor, as it completely conditions his/her tactics during the race. Therefore, within the Tokyo 2020 Weather Project a big amount of data are produced both by collecting data on the sea and by running numerical weather prediction models. All these data are stored into a cloud database.

## 1.3 The motivation

The increasing number of meteorological data available from weather models together with recent developments of technology represent a significant opportunity for the identification of repeatable weather patterns that can support actors working within complex environmental systems. Nevertheless, at the moment, the identification of weather patterns still involves a subjective interpretation from a meteorologist who is linking data coming from numerical weather prediction models, numerical data collected on the field and qualitative signs observed in different weather conditions. This process requires a significant human effort, resulting in a slower analysis of a limited number of data. Moreover, if the area of interest changes, all the manual process should start from the beginning. Automatizing this process would mean spending less time in generating predictions, would permit the analysis of a wider range of meteorological data and would provide procedures that can be reused for different places. Consequently better forecasting could be produced as it would take into account as many parameters as possible and it might be more easily quickly updated.

## 1.4 Objectives

The aim of this work is to find a manner to analyzed automatically this data using **machine learning**, that is a technique that allows a computer to learn from data without being specifically programmed. The goal is to give added value to traditional classification schemes for wind patterns, based on meteorological experience and manual analysis of synoptic weather charts. A methodology based on clustering able to automatically induce wind patterns based on collected data, as well as the characteristic features of these patterns and their evolution through the day, will be developed and

tested. The different automatic clustering that will be tested could also be able to describe different behaviours of the wind in different sailing areas, within the same wind pattern. All these would allow:

- A detailed analysis to determine the representativeness of the wind fields encountered during training and racing period, their frequency of occurrence, timing, rate of evolution, and transition probabilities.
- Consequently, a more thorough prediction of the conditions expected before a sailing race, which is as mentioned a highly valuable information for the sailor.

The system developed will be fed with meteorological data and apply machine learning in the form of *clustering analysis* that consists in grouping together elements that have similarities, to find common pattern of winds to be used to plan the strategy in boat racing. Machine learning indeed fits very well this scope as it requires as many data as possible, and the measurements collected from sensors are many. What will be explained throughout this document is the technique used to find these patterns.

There are multiple applications of different techniques of clustering for finding different kinds of weather patterns. For example, Aran et al. [1] apply PCA analysis in combination with k-means clustering plus Discriminant Analysis to detect weather patterns associated with strong wind events in Catalonia. In this case, in order to compute this classification, the chosen algorithms that will be employed are ***hierarchical clustering*** and ***k-means***. In the literature there are already attempts to take advantages of these two algorithms to analyse winds. In particular the works of Kaufmann and Whiteman [2] and Kaufmann and Weber [3] are two studies where this methodology was applied. In [2] they analyzed wind pattern in the Gran Canyon using data coming from meteorological stations. This paper was indeed the main reference as they proposed solution very suitable for the scope of this work. In spite of that, it is important to emphasise that the wind patterns object of this work aim to be far more specific than theirs, since a race in Olympic sailing takes place in a very small area (around 2-3 nautical miles), and the forecast needed must be far more accurate than a conventional weather forecast: the solution is aiming for the prediction of variations in direction of the wind of 5-10 degrees and variations of 2-5 knots in its intensity during the span of 3-6 hours that the races may last. Furthermore their work was more focused on the meteorological aspect, so this research wanted to go deeper in the machine learning aspect, trying to improve their work using a methodology studied by Surdeanu et al. [4], that similarly to Kaufmann and Whiteman used clustering to group documents, but improving the automation of the algorithm, as the choice of the

right number of clusters after the hierarchical clustering was taken by the algorithm. This two approach will be combined together: finding automatically the optimal value for the number of cluster in hierarchical clustering and apply it in k-means. It should be noted that, although the ultimate goal is to apply this automatic methodology to the actual collected data, at the moment of implementing this work these data were not available yet, since the collection of data in Tokyo started in August 2018 (and will last until 2020). Therefore, our methodology will only be applied to the output from the numerical weather prediction models. High resolution weather models provide wind forecasts for specific areas and specific ranges of time. Although of course it can always be the case that what is forecasted is not what actually happens in the end, it can be a good approximation. The idea is to develop and test the methodology, and in case the results are promising, it will be easy to extrapolate it to whatever geographical area and whatever type of input data, either collected or generated from the weather models.

The project was developed to work independently by the site chosen, and, given that meteorological data of the sites of Tokyo where the races will take place are not yet available, measurements of Barcelona area where used to study, train and check the program.

The program was realized in *Python* chosen firstly for a discreet number of library dedicated to machine learning. These however revealed not to be suitable for the context treated in this work, so the algorithm were written by scratch, while using different libraries for the computational part and the processing of the data.

## 1.5 Structure of the solution

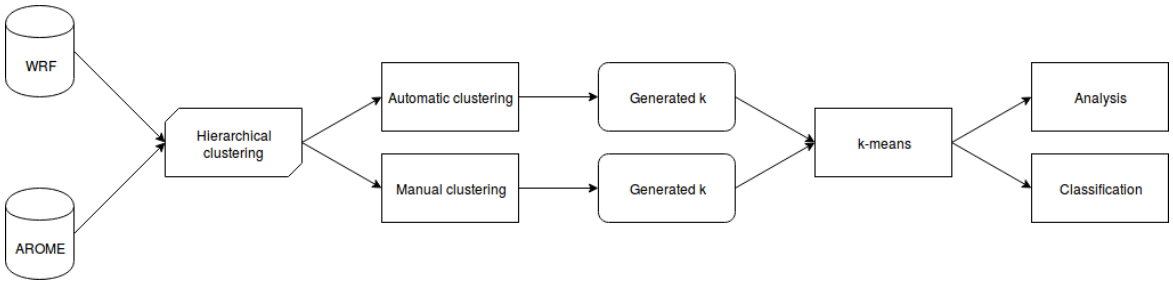



Fig. 1.1 Representation of the components of the solution

In the figure 1.1 it's possible to see how the solution is structured: the available data comes from two different sources of meteorological data, **AROME** and **WRF**.

This two sources came with different file representation but both covered the area of Barcelona and they were analyzed in the same way.

The data was firstly processed by *hierarchical clustering*, in particular in the two different approaches that were mentioned before: **Automatic clustering** corresponds to the work of Surdeanu et al. [4] while the **Manual Clustering** is a plain execution of a hierarchical clustering, as done in Kaufmann and Whiteman [2]. This two different executions both produce $k$, the number of cluster to be used in **k-means**.

Once the classifications are complete, it produces a series of statistics and graphs that are analyzed to check if the resulting classification was able to group wind patterns characterized by similar measurements values and to compare the two different approaches.

## 1.6 Next chapters

In the following chapter 2 the datasets used will be examined, as well as the type of files used to represent them. Next in chapter 3 the theoretical notion of clustering regarding the algorithms used will be explained and, after that, in chapter 4 how they were implemented in this work. Subsequently the chapter 5 is dedicated to the analysis of the results obtained from the data and the algorithms. Lastly the conclusion and the thoughts for a further work at chapter 6 close this thesis.

# Chapter 2

# The meteorological data

## 2.1 Data collection

The first step in the process that leads to the weather forecasting is the data collection. This is usually done with satellites or weather stations equipped with different instruments, like barometer, thermometer, anemometer and more. These can be stations on the ground, or, as this will be the scope of this work, dedicated buoys to measure weather data in the sea.

## 2.2 Types of data

As we said the meteorological stations are furnished with many instruments that check various air parameters. In this work different types of parameters, know by the general public, are used:

- temperature
- humidity
- pressure
- total precipitation

In addition to these, there are two parameters that are the most important and the ones used in the algorithm calculations, that are related to the wind. They are the wind **u** and **v** components that are the mathematical representation of the wind flow as a vector.

The wind has a speed and a direction. These two elements can be completely defined by a vector.

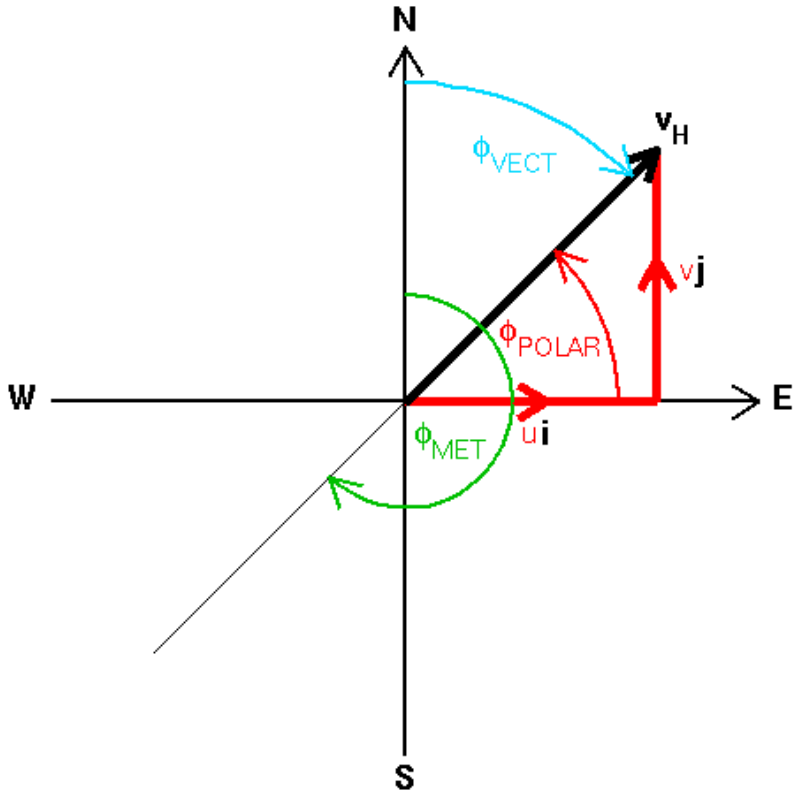



Fig. 2.1 Vector representation of the wind flow

From the figure 2.1 [5] is possible to see the vector defined by a wind flow $v_H$. The components **u** and **v** (also called *eastward direction* and *northward direction*) are the projections of the vector on the axis. The representation of the direction however is not simply the angle $\phi_{polar}$ generated by the vector with the x axis (corresponding to $E$ in the figure 2.1). There are two conventions to represent the direction: one is the *wind vector azimuth*, that is the direction where the wind is blowing, corresponding to the angle $\phi_{VECT}$ in the figure. The other is the *meteorological wind direction*, i.e the direction from which the wind is blowing, represented by the green arch $\phi_{MET}$. It is important to keep in mind this representation in the section of calculation, otherwise this could lead to wrong interpretation of the meteorological data.

In the rest of this document, the direction will be always considered as *meteorological wind direction*.

## 2.3 Data sources

### 2.3.1 Numerical Weather Prediction

Numerical Weather Prediction, or NWP, [6, 7] is a method of weather forecasting that employs mathematical models of the atmosphere that use current weather conditions to elaborate forecasting. A NWP is calculated with the help of a computer and there are many distinct models available that differ based on the atmospheric processes they are applied to or the part of the world that they analyze. The current weather observations are the input of numerical computer models that calculate outputs of many meteorological parameters, like temperature, pressure and many more, thanks to a process called data assimilation. So both the meteorological measurements and the numerical computer models have great importance in the forecasting.

The numerical models used consist in equation of fluid dynamics and thermodynamics that describe the atmosphere behaviour and predict it. Such equations are complex to solve and require to be simulate on super computers. This is a reason because meteorological forecast can predict weather up to six days, and also because the equations are nonlinear partial differential that cannot be solved exactly but the results obtained are approximate solutions, and the error grows with time.

The data that has been used for this work comes from two different weather models: **WRF** and **AROME**.

#### 2.3.1.1 WRF

The Weather Research and Forecasting (WRF) [8] Model is a mesoscale numerical weather prediction system developed starting in the latter part of the 1990's, designed for atmospheric research and forecasting application. It was developed in the US by many research entities and universities and today is maintained by National Center for Atmospheric Research (NCAR). It is composed by two dynamical cores, a data assimilation core and a parallel and extensible software architecture. It can produce simulations using actual atmospheric conditions or idealized conditions. WRF can count on a large developers and users community and it is widely used for meteorological bulletin as well by researchers and laboratories.

This is a high resolution model that serves applications across scales from tens of meters to kilometers and is not freely available to the public. For this reason this meteorological data was provided by **Meteocat**, the meteorological service of Catalonia. In this project they collaborated providing the data and the know-how on how to deal with different kind of files. From this model they furnished the files with

the six weather parameters mentioned earlier needed in this project, in csv (comma separated values) files.

The data is contained in csv files that cover all month of March and April 2018. Each file represented a precise time and date of the period and contained the value of one measurement over different locations of the area of Catalonia. As said in 2.2 the types of data used are 6. That means that for every timestamp there were 6 different files each containing one measurement.

Table 2.1 Example of WRF file

| | | |
|---|---|---|
| 41.275 | -1.225 | 6.787735 |
| 41.275 | -1.175 | 5.787735 |
| 41.275 | -1.125 | 5.58461 |
| 41.275 | -1.075 | 5.287735 |
| 41.275 | -1.025 | 5.17836 |

The table 2.1 shows an extract of a csv file: the first column is the latitude, the second is the longitude and the last one is the measurement. Using this type of files was particular easy: first of all it is a common and spread text file format where values are separated by commas, or other special characters, stored in tables. It is also easy to read with just a spreadsheet program, like Microsoft Excel. These files are small in size and each of them was less than 30 KB, and, even if each of the six parameters was saved in a single file, the whole data set is, yes composed by more than 8000 files, but still occupied a total space in local memory of 240 MB, a huge difference in what will be seen with the AROME dataset. Lastly operating with csv files in Python is straightforward: the function *genfromtxt* from numpy, a powerful library of Python that allows to manipulate data and mathematical functions, read the all file row by row, and save it in an array.

#### 2.3.1.2 AROME

AROME, Applications of Research to Operations at MEsoscale, [9] is a small scale numerical weather prediction model, mantained by Meteo-France, designed for short term forecasting, in particular for severe atmospheric phenomenons, like storms in the Mediterranean. It covers all France territory and waters and most of the Spain.

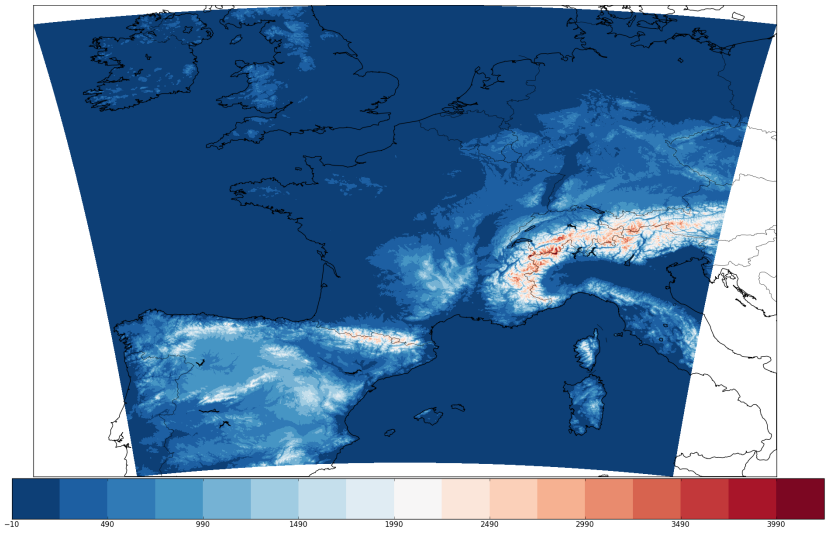



Fig. 2.2 The area covered by AROME

AROME collects data from radar networks to produce hourly based models with a high resolution. It is used to produce five days weather predictions but it has also proven to be useful with severe precipitations.

This data was provided instead by TriM that was the main partner in this project that along data contributed with knowledge and collaborated with Meteocat too.

The original data were saved in GeoTIFF files, one for each parameter. TIFF, Tagged Image File Format, is a tag based file format designed to store and share raster images. GeoTIFF is an extension compliant with TIFF specifications used to store georeferenced information thanks to certain TIFF tags.

It was planned to merge all the measurements of one day in a single file and, as a results of the meetings and advice from Meteocat, there are two files suitable for this: *GRIB* and *netCDF*.

**GRIB files** One choice to store the meteorologic data was *GRIB* (GRIdded Binary or General Regularly-distributed Information in Binary form) files [10, 11]. This particular file format is commonly used in meteorology to store weather data. Created by the World Meteorological Organization (WMO) it is a collection of records of data, in the form of tables, thought to transfer volumes of gridded data efficiently. Data packed in a GRIB are more compact than a text oriented file, that means smaller files to store large amounts of meteorological data. There are two versions of GRIB currently used, GRIB1 and GRIB2. They differs for additions of new parameters or

more precise definitions of the existing ones. A record of a GRIB file represents a parameter described by gridded values or coefficients. In this case the parameters represent the u and v values, temperature, humidity, pressure and total precipitations.

To operate with this kind of files there are different libraries and programs. The most used is *wgrib*, available as a bash command in Linux. The usage is quite simple, it allows to decode GRIB files and read their contents or convert them into other files.

Despite its spread, GRIB files presents some disadvantages that prevented to be used in this project. As said before, there are two main versions of GRIB and to read GRIB2 is necessary to use *wgrib2*. It requires many dependencies from other libraries to be installed therefore it was hard to configure everything correctly. Furthermore the parameters can be set differently as well as the grid of coordinates used. Lastly the output decoded from a GRIB file using wgrib is not easy to interpret. For example the output from a simple wgrib command of a wind measurement file is the following:

```
$ wgrib Lyon-Baleares.grb
1:0:d=18092018:UGRD:kpds5=33:kpds6=105:kpds7=10:TR=0:
      P1=6:P2=0:TimeU=1:10 m above gnd:6 hr fcst:NAve=0
2:51186:d=18092018:VGRD:kpds5=34:kpds6=105:kpds7=10:TR=0:
      P1=6:P2=0:TimeU=1:10 m above gnd:6 hr fcst:NAve=0
3:102372:d=18092018:UGRD:kpds5=33:kpds6=105:kpds7=10:TR=0:
      P1=9:P2=0:TimeU=1:10 m above gnd:9 hr fcst:NAve=0
4:153558:d=18092018:VGRD:kpds5=34:kpds6=105:kpds7=10:TR=0:
      P1=9:P2=0:TimeU=1:10 m above gnd:9 hr fcst:NAve=0
```

This an extract of the command result. Firstly it can be noted that it is not very easy to understand. It includes information on the data like the byte offset, that is not vital to understand the content, while other data are stored in the values kpds: kpds5 with value 33 states that it is a u-component of wind; kpds6=105 that it is measured at a certain value above the ground and lastly kpsd7=10 that the measurement took place every 3 hours. However all this information present in kpsd values can be understood only checking the tables in the documentation of grib files that explain what each value represents. This is clearly not very immediate to understand and requires time to research the meaning of a value. To overcome this problem, it was agreed that all AROME files used in the project should have been converted into *NetCDF* files.

**netCDF** netCDF (Network Common Data Form) [12] is another format to store array-oriented scientific data in a portable and self-describing file, that means that this kind of file can be accessed on different platforms, regardlessly how they represent

integers, floats and characters, and also that the dataset includes a description of the data that is stored, like the units of measurement.

This type of file is developed and maintained by Unidata that is part of the University Corporation for Atmospheric Research (UCAR), indeed there are many tools provided by Unidata to work with geoscience data. As said the data are in form of arrays and this include arrays of dimension $n > 1$, and the data used in this project are an array of dimension 3, latitude, longitude and time. This can be seen as a 3-D matrix composed by 24, as the daily hour, 2-D matrices with the dimensions of the area covered by the meteorological data, each element of the matrix containing the measurement of the parameter.

This file has evolved through the years and there are different versions released over the years. Until version 3.6.0 the versions of netCDF employed one binary format. These are referred as classical format. After 3.6.0 a 64-bit offset format was adopted allowing to use file bigger than 2 GiB. The last version 4.0.0, called netCDF-4, started using another binary format HDF5, that is a spread data model for storing and managing data, capable of supporting a wide variety of datatypes. It is designed to be flexible and efficient thus is portable and extensible.

The new versions allowed the netCDF file to store a larger amount of data with the 64-bit offset update and with the netCDF-4 format it was possible to use more complex representation of data, like groups, nested trees and variable lenght array.

Despite the improvements of the newer versions, the classic format is the one that grants the maximum compatibility as the usage of later versions requires the interfaces and programs to be updated. So in this project the format for the netCDF files is classic format.

This kind of file can be read easily via a Unix command, *ncdump*. In the following example we can see an extract of the structure of the variables of one AROME file.

```
$ ncdump -h AROME_2018-04-06.nc
dimensions:
    time = UNLIMITED ; // (48 currently)
    lat = 561 ;
    lon = 590 ;
variables:
    double Band1(time, lat, lon) ;
        Band1:long_name = "eastward_wind" ;
        Band1:_FillValue = 9.96920996838687e+36 ;
        Band1:grid_mapping = "crs" ;
```

```
double Band2(time, lat, lon) ;
    Band2:long_name = "northward_wind" ;
    Band2:_FillValue = 9.96920996838687e+36 ;
```

A classic format netCDF is composed by two parts, the header and the data. The *-h* argument of ncdump shows the header. This describes the type of content and how it is represented. It is possible to see that the dimensions that describe the variables are 3, time, latitude and longitude. This means that data vary by these 3 dimensions. Time in particular is defined as *unlimited* that is used for dimensions that can be extended, in cases when the total length is not know or it is necessary to add more data. On the other hand latitude and longitude are well defined because a specific area is covered.

Speaking of the variables, two of the total 6 are present in the extract. They are a 3 dimensions array, represented by double values defined by the three dimensions and are respectively the u and v component. It is possible to specify a default value when some are missing, a name for the variable and the mapping of the grid used and the real coordinates.

Using this kind of file within Python was easy, it was just necessary to install a module, simply called *netCDF4* that uses a class called Dataset. Accessing the data was relatively simple too, accessed as a 4 dimensions array, one dimension for the variable name (i.e. Band1, Band2, etc.) and 3 for the dimensions time, latitude and longitude.

Using such files revealed easy and handier than GRIB files, nevertheless this type of format had a con, that is its size. The files used to store AROME data were on average 700 MB for a total of 40 GB for the complete dataset, and this affected the program performances on the loading of the data, a significant difference with the csv file of WRF.

A useful program used during this work to inspect and check NetCDF file was PanoplyJ, developed by NASA. Thanks to this program was easy to check the types of variables stored in the file and how they were structured.

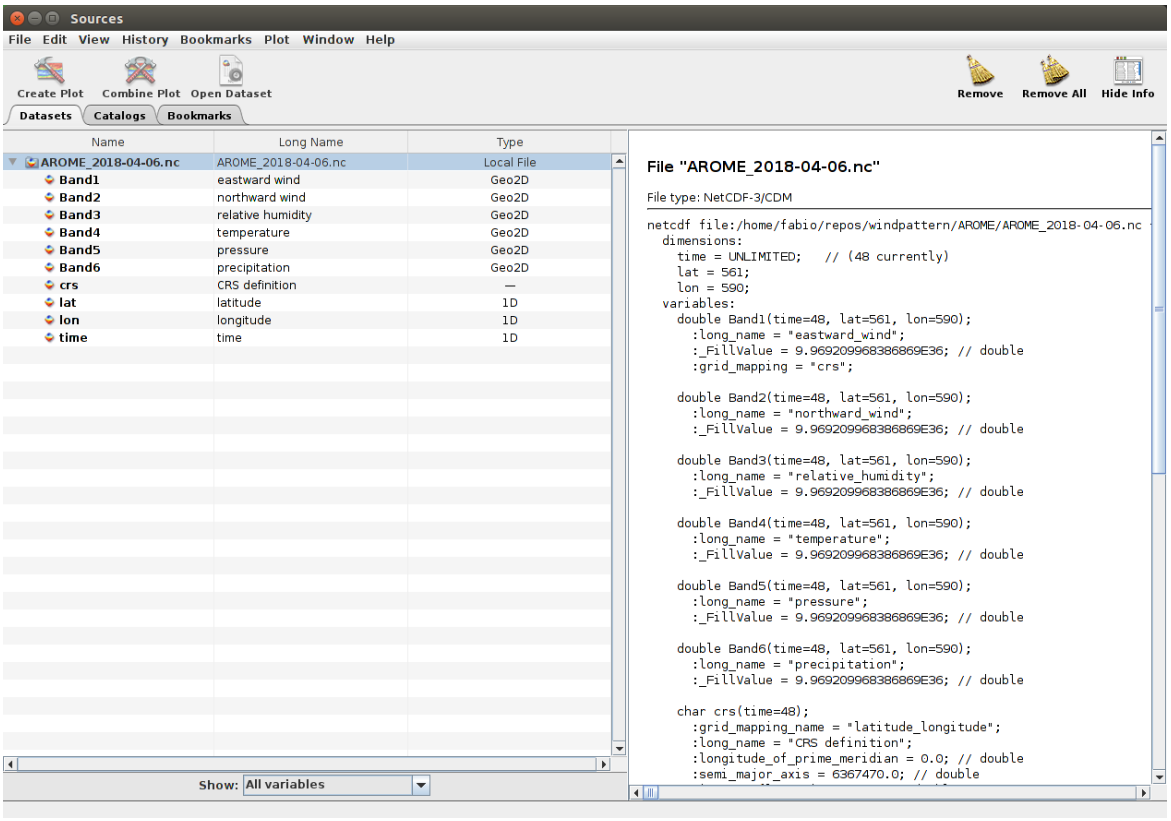



Fig. 2.3 Screenshot from PanoplyJ

From the screenshot of PanoplyJ in figure 2.3 we can understand better the composition of these files. On the right there is a pane with the same information given by ncdump, regarding the dimensions and variables. On the left pane all the variables and dimensions are single elements in a table, with additional information. Long names and the type of data are clearly expressed; it is interesting to note that the different variables Bands are Geo2D type. Indeed they are a collection of 2D matrices georeferenced data, in this case 48.

In PanoplyJ, is also possible to plot them on a map and have a visual representation of the values.

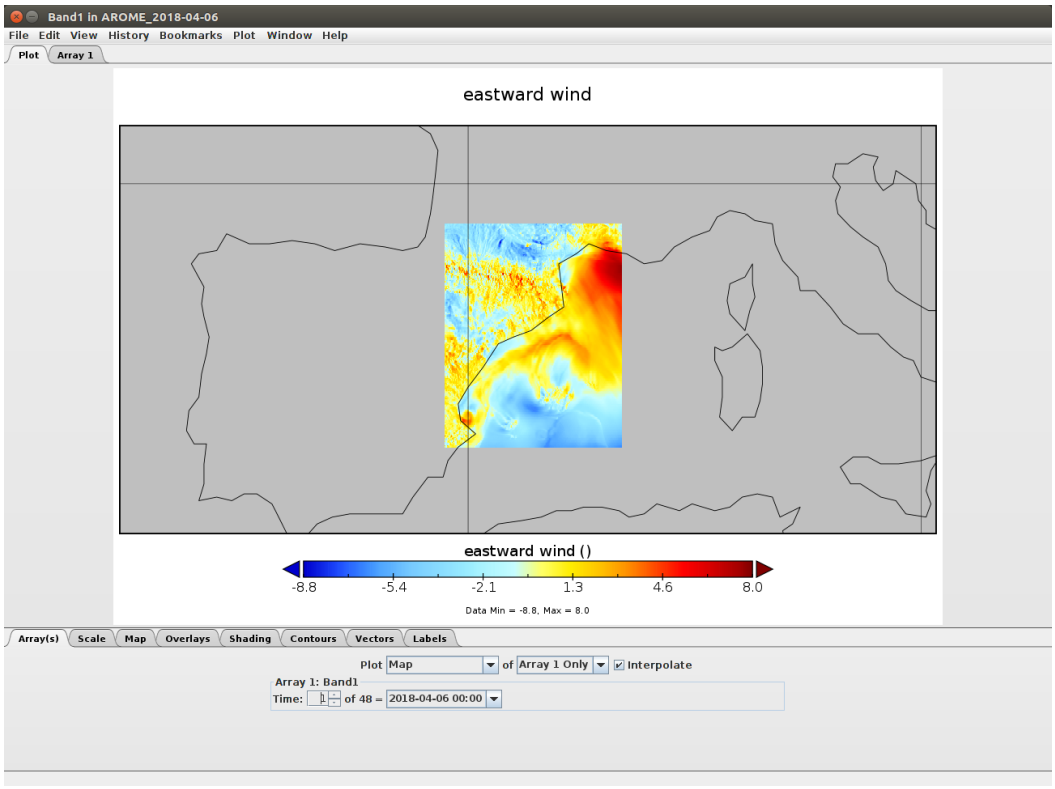



Fig. 2.4 Georeferenced Plot

Lastly it is possible to look at the values of this variables.

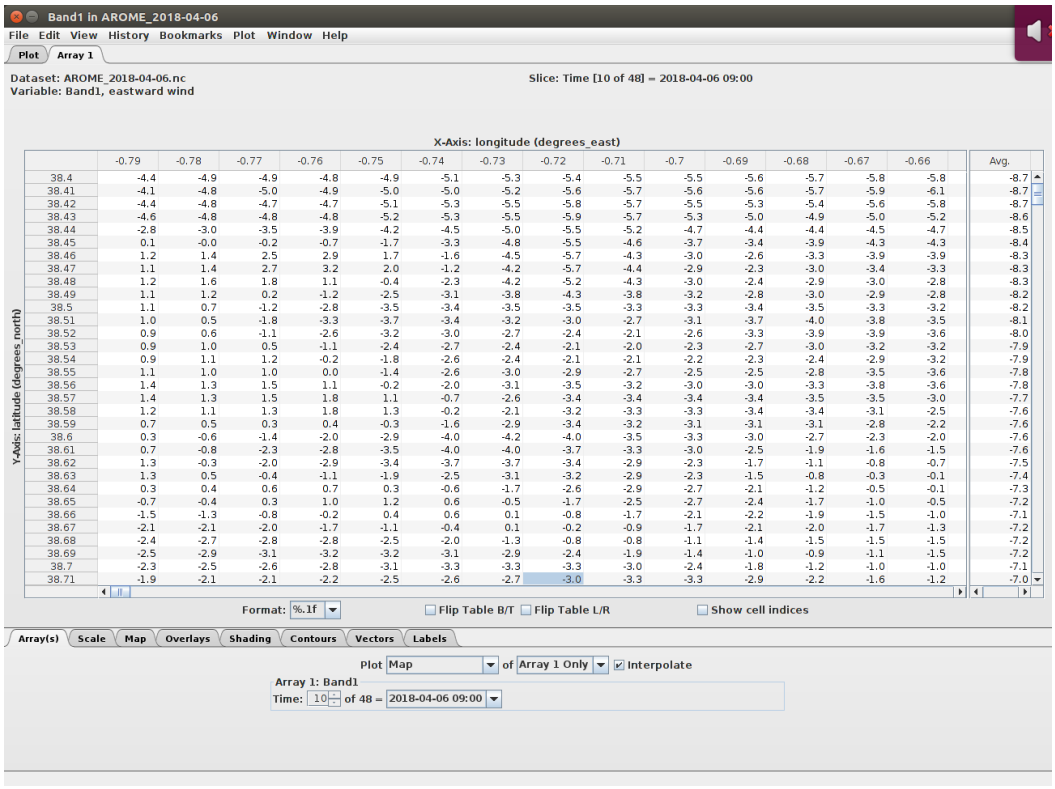



Fig. 2.5 Representation of the values stored in the NetCDF file

Every cell is the value of the measurement in a certain place given by the coordinates in the row and column header. The whole day is represented by 24 of this tables (even 48 as it is forecast also the following day) that can be selected in lower part of the window of figure 2.5. This table is exactly what will be loaded in Python.

During the work on the project however, it surfaced that some values of the temperature of the days of May were wrong, indeed they had values of more than 10000. Checking the files with Panoply and talking with TriM revealed that there had been a mismatch during the conversion into netCDF. Firstly a workaround was used to exclude this wrong values in the code of the program, then the files were fixed, as those mismatched values referred instead to pressure. This has affected marginally the initial results during test because, as said before, the main values used by the algorithm are u and v. Anyway in the final results, this mistake was corrected.

Choosing netCDF as the format to operate with revealed the best choice, weighting the compromises between the ease of use and the performance.

### 2.3.2 Area of measurements

This calculation are aimed for the area of Tokyo that is where the sailboat races will take place. However measurements from Japan are not yet available, so to test the program it was chosen the bay of Barcelona, due to both the availability of high resolution data and the knowledge of the area from the supervisor of this thesis and TriM. This knowledge will be very valuable when analyzing the usefulness and coherence of the results obtained.

As in Kaufmann and Whiteman [2] there were weather stations measuring wind data, in this project 16 coordinates of the two weather model were taken as to simulate measuring station. These comprehend both points on the land and in the sea.

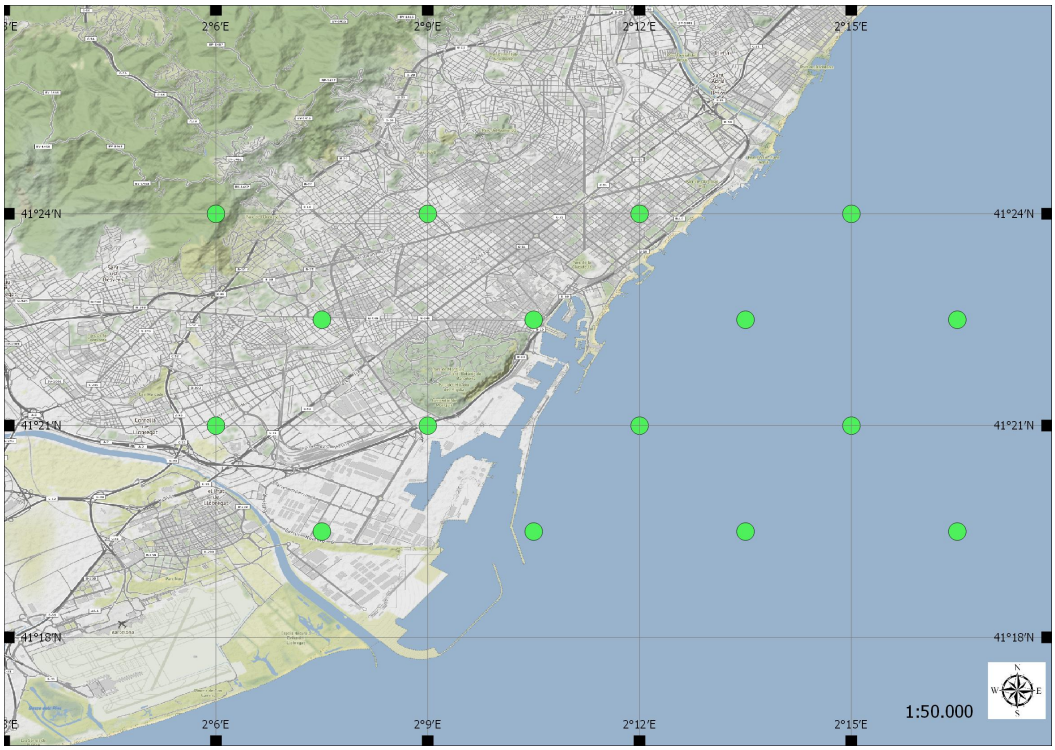



Fig. 2.6 The 16 points chosen for WRF

As it possible to see in figure 2.6 half of the points are on the land while the other in the sea. In this way it was possible to test the program with winds that characterise different location.

The points chosen for AROME are quite similar but they differ slightly just because the WRF has a higher resolution, although they can be considered corresponding.

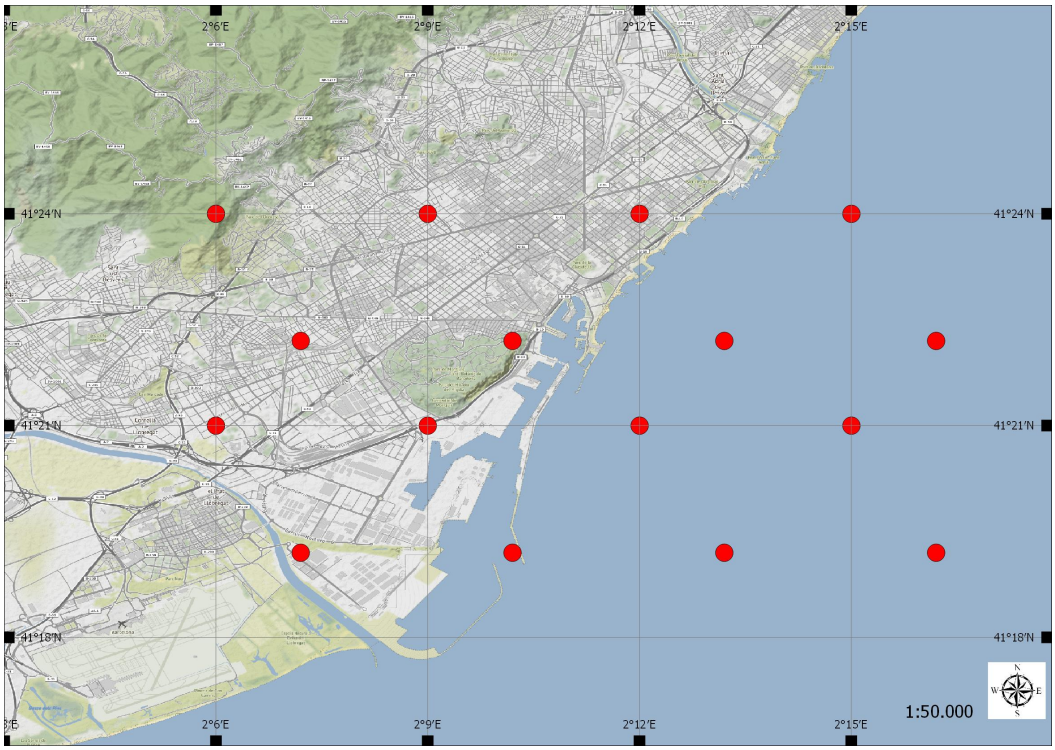



Fig. 2.7 The 16 points chosen for AROME

# Chapter 3

# Clustering

## 3.1 Machine Learning

There are many definitions of *machine learning*. The most famous is the one gave by Arthur Samuel, pioneer in machine learning in 1959: ″Machine learning is the field of study that gives computers the ability to learn without being specifically programmed”.

There is also another definition, more formal, gave by Tom Mitchell: "a computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E" [13].

In some way the aim is to imitate how the human brain learns, using statistical techniques and algorithms.

There are many and different learning algorithms, but they can be grouped in two main categories depending on how the algorithm learns: **supervised** and **unsupervised learning**.

In **supervised learning**, given a dataset, it is already know what is the correct output, so every measurement has a corresponding response. It is like teaching a computer what is right and what is wrong, giving it examples on how it should answer with specific inputs and learn the rule that maps inputs with correct outputs. Examples of supervised learning are *regression*, where input is mapped on a continuous output, and classification, where the predicted outputs are discrete, that means starting from inputs find the corresponding outputs in a series of classes or categories [14].

On the other hand in **unsupervised learning** the dataset has no correct output, so it is not possible to teach the computer how a correct answer should be, it has to learn it by itself. In this case it is harder to analyse the results of a learning algorithm since the correctness of the output is unknown to the programmer too and the goal is

find pattern in the data. Nevertheless the advantage of unsupervised learning algorithm is that it is easier to find unlabelled data rather than label ed data, that requires human intervention [15]. Examples of unsupervised algorithm are *clustering*, *dimensionality reduction* and *neural networks*.

*Dimensionality reduction* consists in reducing the dimension of the data that need to be analysed to optimised computing time and costs. It assumes that data is redundant and that it can be represented with only a fraction of it [16] [17]. There are two popular algorithm to reduce the dimensionality:

- *Principal Component Analysis* (*PCA*), that produces a low-dimensionality of a dataset, finding a linear combination that gives the maximum variance;
- *Singular-Value Decomposition* (*SVD*), that allows to represent data as a product of 3 smaller matrices.

*Deep learning* is a relatively new approach, based on the use of *neural networks*, that tries to imitates how brain cells work, decomposing the problem in smaller task adopting elements called perceptrons that calculate output using functions and weights, composed in single or multiple layers. [18, 19]

Lastly clustering, that is the topic of this work, will be examined more in details.

## 3.2 Fundamentals of clustering

Clustering is a machine learning technique where the aim is to find clusters, or groups, in a dataset. As said it is part of **unsupervised learning**, that means the input data is not labelled by a supervisor nor there is any indication of what defines a good value or a bad value.

Clustering the observations of a dataset means partition them in groups of objects that are similar to each other, while clustering objects that are different in other groups. ″Similar” however is a generic concept and depend on the domain of application. [14]

As an example it can be taken a series of clinical data from cancer patients. The sample of the dataset are expected to be heterogeneous, depending on the patient data. However certain aspects or subgroups of them could be similar between some, defining a certain type of cancer or other common characteristics. Clustering these data could help finding this common subgroups without having to classify them previously.

The field of application of this kind of method are numerous, thanks to the fact that data does not need to be preprocessed. As said they can be used with clinical data, but also in marketing to group clients that have common characteristic who can

be target of an effective advertising campaign or to drive them to the most suitable purchase.

Also they can cluster documents, for example newspaper articles, dividing them in bag of words, and can be categorized by type (sport, economics, politics, etc.)

There are many clustering algorithms: **hierarchical clustering** and **k-means** are among the most spread and are the ones adopted in this project, but there are other important clustering algorithms such as *DBSCAN* and *Expectation-Maximization*, which have been discarded because they do not apply to our specific methodology, as described in section 1.5. However, they will be briefly mentioned as well due to their importance in the clustering group.

*DBSCAN* [20, 21] is a density-based cluster algorithm that is a class of clustering where clusters are defined as area of higher density compared to the rest of the data, while sparse object are considered noise. With this kind of clustering is possible to discover clusters of arbitrary shape. DBSCAN in particular is based on the concept of density-reachability that defines cluster as a maximal set of density-connected object. Objects are divided in three categories:

- core, that are the objects which have a minimum number, defined as a parameter, of other objects within a distance threshold;
- border objects have less elements than the minimum number in their neighbour, but are in the neighbour of a core object;
- outlier or noise object, that are neither core nor border objects.

So a cluster is composed of core and border points.

DBSCAN pros are that it is able to ignore noise and handle cluster of different shapes and sizes. However it needs area where the density is lower to recognise the border of a cluster.

The distribution of objects can be also hypothesised to be a mixture of distributions where every group is generated by a single distribution but all the groups together appear as generated by a unique one.

The goal of *Expectation-Maximisation* (*EM*) [4, 22–24] is to find the original distributions. To simplify the process they are considered all the same type, usually Gaussian. EM tries to find the parameters of the distributions calculating the log-likelihood, a simplification of the maximum likelihood estimation. EM executes iteratively the two steps Expectation and Maximisation: in the expectation phase the expected value of the log-likelihood is calculated, while in the maximisation phase

the expected value previously calculated is maximised to determine the distribution parameters. The process is repeated until convergence, that assured since the algorithm maximises the likelihood at every step.

## 3.3 Hierarchical Clustering

Hierarchical Clustering [14, 25] is one type of clustering that uses a distance function to establish if two objects are similar and thus be in the same cluster. Usually Euclidean distance is used when attributes have the same scale

$$d(x, y) = \sqrt[n]{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{3.1}$$

There are also other distance functions, like correlated-base distance. It considers two observations similar if their features are correlated, even if they are far with Euclidean distance. For example customers that buy few items can be clustered together, but maybe their shopping preferences might be very different; correlation-base distance would pair people with similar preferences independently of the number of purchases.

Hierarchical clustering can be *bottom-up* or *agglomerative*, i.e. it starts with N clusters, each containing only one element and each step of the clustering consists in merging the two most similar, until only one cluster is obtained. The opposite is the *top-down* or *divisive* clustering that starts from a unique group and divide it until every observation is in one group. This work will use the agglomerative version, that is also the most common one, starting from a group for every wind observation.

The result of Hierarchical clustering is a *dendrogram*, a tree-based representation of the observations. The leaves represent all the observation and the tree shows how they are merged.

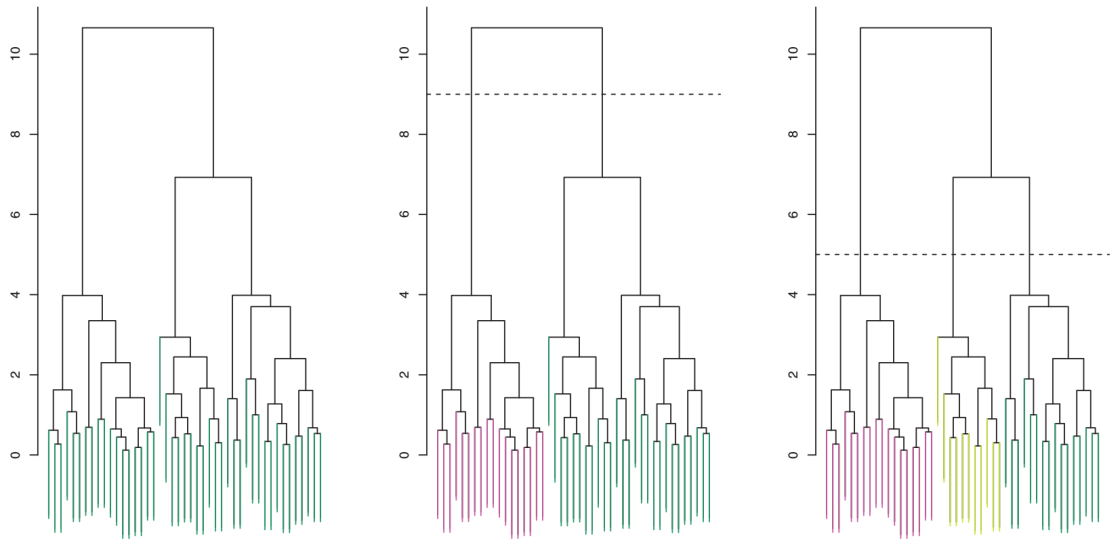



Fig. 3.1 Examples of dendrograms

Figure 3.1[14] shows some examples of dendrograms. From the y axis we can see how distant two elements are: the shorter vertical line that connects two observations is, the more similar the observations are. And moving up the tree gives indication of the order of the merging, because the two closest elements are the ones chosen to be joined, so the first horizontal line that connects two elements is the first cluster to be created.

The dashed lines in the second and third dendrograms is the height at which the dendrograms are cut, respectively 9 and 5. The number of intersection with the vertical lines represent the number of clusters that will be formed. Cutting the dendrogram at a height of 9 gives two clusters that contain respectively the elements in red and green. The height of 5 gives instead three final clusters. The height of the cut controls the number of cluster obtained.

The downside of this technique is the choice of the height, so the choice of the number of clusters. This is usually done by observing the clusters and picking an arbitrary height. Clearly this is not the best solution, as it could require more attempts or either is not easy to understand for complex data if a cluster has effectively similar observations.

### 3.3.1 The algorithm

The algorithm of this method, illustrated in Algorithm 1, is really easy. It first needs a dissimilarity measure to sort the observations. Usually for simple elements Euclidean

distance is the common choice. The algorithm works iteratively, starting from the leaves of the dendrogram where each of the observation has its own cluster. At each step the two most similar groups are merged, so after there are $n-1$ cluster. At the next step they become $n-2$ and so on, until all of them are grouped in a unique cluster.

During the execution the algorithm merges the two closest clusters it finds. This concept is straightforward when two single elements are considered, but for clusters with more than one element is necessary to choose a point that will be used to calculate the distance. So to extend the notion of distance to group of observations, the linkage function was introduced, that calculates the distance between arbitrary subsets of the dataset. The most common types are *single*, *complete*, *average* and *centroid* linkage.

Table 3.1 Types of linkage

| Linkage | Description |
|---|---|
| Single linkage | Computes all the possible distances between the elements of cluster A and cluster B and chooses the *shortest* distance of all. |
| Complete Linkage | Computes all the possible distances between the elements of cluster A and cluster B and chooses the *largest* distance. |
| Average Linkage | Computes all the possible distances between the elements of cluster A and cluster B and calculates the *average* of them. |
| Centroid Linkage | The distance is the point distance between the means, i.e. the *centroids*, of cluster A and B. |

Mathematically they can be expressed as follows

$$L_{Single}(A,B) = \min_{x \in A, y \in B} d(x,y) \tag{3.2}$$

$$L_{Complete}(A,B) = \max_{x \in A, y \in B} d(x,y) \tag{3.3}$$

$$L_{Average}(A, B) = \frac{\sum_{x \in A, y \in B} d(x, y)}{|A| \cdot |B|} \tag{3.4}$$

$$L_{Centroid}(A, B) = d\left(\frac{\sum_{x \in A} x}{|A|}, \frac{\sum_{y \in B} y}{|B|}\right) \tag{3.5}$$

Complete and average linkage are preferred over single linkage as they produce a more balanced dendrogram. Moreover average and centroid linkage offer an advantage because take into account the shape of the cluster, differently than single and complete linkage that compute the distance between points.

**Algorithm 1:** Hierarchical agglomerative clustering

**input** : dataset $D$, linkage function
**output** : The dendrogram representing the clustering

Initialization of data: one single cluster per every object;
Compute the distance matrix, a squared matrix representing the distance between each cluster using the chosen distance function;
**while** *the number of cluster > 1* **do**
    find the pair X, Y of closest cluster;
    merge them;
    Update the distance matrix removing the rows and columns of X and Y and adding a new row and a new column with the distances of the new cluster;
**end**
**return** the dendrogram formed;

## 3.4 K-means

The other clustering method taken in exam is *k-means* [14, 25, 26]. This simple algorithm partitions the data in K non-overlapping clusters. It requires the number k of clusters to be firstly specified, then it proceeds iteratively to assign all the elements to one of the k clusters. K-means is based on the idea that in a good cluster the within-cluster distance (or variance) is as small as possible. The within distance is a measure that indicates how the elements that belong to a cluster differ from each other: if it is small it means that the elements within the cluster are all very similar.

$$\underset{C_1, \ldots, C_K}{\text{minimize}} \left\{ \sum_{l=1}^{K} W(C_l) \right\} \tag{3.6}$$

The formula means that for all partitions $C_i$ we seek to minimise the within distance.

The within-cluster distance has to be defined and, similarly to hierarchical clustering, this can be the squared Euclidean distance.

$$W(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 \tag{3.7}$$

is sum of the pairwise squared Euclidean distances of the objects of the k-th cluster, divided by the number of elements in the cluster, denoted by $|C_k|$.

Using Euclidean distance implies that the clusters generated will be hyperspheres. The most used algorithm, known as Lloyd's algorithm, is an heuristic algorithm of the k-means problem, which is NP-complete, which means that it cannot find a global optimum in an efficient way. It can be generalised as follows.

**Algorithm 2:** K-means algorithm

**input :** dataset $D$, value of K
**output :** K clusters

Randomly initialise K vectors $C_1$, ..., $C_K$;
**repeat**
  **foreach** *cluster* **do**
    compute the cluster centroid as the mean of cluster elements;
  **end**
  assign each observation of D to the cluster of the closest centroid;
**until** *no changes in* $C_1$, ..., $C_K$;
**return** the clusters $C_1$, ..., $C_K$;

The algorithm partitions the data in initial random clusters and then at every iteration assigns the elements to the cluster of the closest centroid, recalculating the centroids for every cluster.

K-means algorithm is guaranteed to decrease the within-distance of clusters at every step, thus the clustering always improves until there are no more changes. When it stops it means that it has reached a local optimum, but that does not assure that it is the best solution possible. For this reason it suggested to run the algorithm different times with different or randomized initial clusters.

The disadvantages of k-means regard the choice of k. It's not easy to establish a priori the best value, so it requires multiple runs to choose the outcome with the

minimum within-distance. Of course the number of k affects also the performance as larger values require more computing time.

### 3.4.1 K-means variant

In this work a variant of the k-means was adopted, as done in Kaufmann and Whiteman [2], the *Wishard's variant*[27], that aims to remove outliers from the clusters. As mentioned previously, outliers are observation point that are distant from the others. They can be due to measurement errors and cause problems in the calculation, so it is better to discard them.

The key element of this variation is the *threshold* that discriminates between outliers and regular elements.

The algorithm is the following:

**Algorithm 3:** K-means algorithm - Wishard's variant

**input** : dataset $D$, value of K, THRESHOLD, MINSIZE, MAXITER, MINC
**output** : MINC clusters

Randomly initialize K vectors $C_1$, ..., $C_K$;
**repeat**
  **repeat**
    **foreach** *cluster* **do**
      compute the cluster centroid as the mean of cluster elements;
    **end**
    **foreach** *element x in D* **do**
      compute the distance of x from the centroids;
      **if** *distance of x > THRESHOLD* **then**
        assign x in the outliers residue;
      **end**
      **else**
        **if** *x is in residue or in another cluster* **then**
          assign x to the cluster of the closest centroid;
        **end**
      **end**
    **end**
    **if** *size of any cluster* $C_j$ < *MINSIZE* **then**
      assign cluster to the residue;
    **end**
  **until** *no changes in* $C_1$, ..., $C_K$ *OR number of cycles > MAXIT*;
  calculate pairwise similarities between clusters and merge the two most similar;
**until** *the number of cluster == MINC*;
**return** the clusters;

This algorithm maintains the k-means principles, but for every element checks if it is an outlier examining their distance from the centroids, and if it is greater than the threshold, it is moved in the outliers group. Nevertheless it can exit the group if the clusters and their centroids change such that its distance from one centroid becomes smaller than the threshold: it is moved out from the outliers and admitted back as regular elements. The MINSIZE defines the size, usually small, that characterize the minimum cluster dimension to be considered regular, otherwise it would represent a

group of outliers. The MINC assures that a certain number of clusters will be returned since small clusters are marked as outliers and similar ones are merged, operations that reduce the total number of clusters.

This algorithm is thought to construct only the most likely partitions. In this method true outliers are likely to be all assigned to the residue. However during the execution elements can enter and exit the residue due to the changes in the clusters, for this reason there is the MAXIT parameters to prevent infinite loops.

In the implementation of this project it was planned to consider the MINSIZE = 1, thus every cluster with only one element that could not join with other elements was discarded as outlier. However it was not enforced the merging of the two most similar clusters as it was intention to preserve the input clusters configuration.

## 3.5 Distance measure

In the previous sections, talking about the distance used to measure similarity between elements, the Euclidean distance was indicated as the prevailing measure used. Nevertheless in this work the Euclidean distance was not suitable for the kind of data used. The u and v component, explained in section 2.2, are the two parameters that determine the similarities of wind flows, but it is necessary to take into account the timestamps and the location of the measurements.

In order to consider all these variables, Kaufmann and Whiteman [2] introduced a specific distance measure for wind patterns:

$$d_{ab} = \frac{1}{N_{ab}} \sum_{j=1}^{N_{ab}} \sqrt{(\tilde{u}_{aj} - \tilde{u}_{bj})^2 + (\tilde{v}_{aj} - \tilde{v}_{bj})^2} \tag{3.8}$$

that measures the distance between two timestamps $a$ and $b$, where $N_{ab}$ is the total number of sites that are available at both times $a$ and $b$, $j$ is the current location and $\tilde{u}$ and $\tilde{v}$ are the u and v values normalized.

## 3.6 Automatic and manual clustering

As said in the introduction in this project two different approach were used to produce a clustering of the wind patterns, both based on the two clustering methods explained before.

One consists in firstly analyze data with hierarchical clustering, choosing manually the number of clusters and refine the classification with k-means, since it allows elements to move back and forth to the best cluster. This method is based on the work of Kaufmann and Whiteman [2] and it is the object of comparison with another method adopted by Surdeanu, Turmo, and Ageno [4], that, similarly to Kaufmann, uses two clustering algorithms, hierarchical clustering and Expectation-Maximization to cluster a collection of documents. The interesting part in this paper is that the number of cluster resulting from the hierarchical clustering is chosen automatically by the algorithm and it does not require the human intervention. So, the very interest of this work will be to compare these two methods, using for both hierarchical clustering and k-means and analyze the results to understand if the automatic clustering is able to deliver an outcome that is meaningful for the scope of this work and if can perform even better than the manual clustering.

### 3.6.1 Manual clustering

From now on the technique used in Kaufmann and Whiteman [2] will be referred as *manual clustering*. As briefly explained before it adopts both hierarchical clustering and k-means. The form of hierarchical clustering used is complete linkage since Kaufmann and Whiteman found that it is more suitable in classifying wind patterns and it creates clusters that are more balanced in size. The motivation in using both the two clustering algorithms is the effect of outliers: they extends the boundary of a cluster away from its correct mathematical center, thus producing incorrect results, and in hierarchical clustering objects assigned to a cluster cannot move to another one during the analysis. On the other hand k-means allows elements to change cluster and they can be assigned to a more relevant cluster.

However k-means still needs the number of clusters k and it cannot be decided previously because there are no information on the number of wind patterns that can be found. To solve this inconvenient, the input to k-means are the clusters formed by the hierarchical. This does not completely solve the inconvenient of choosing a proper number of clusters, because also in hierarchical clustering has to be decided, starting from the dendrogram. Kaufmann and Whiteman employed a simple method for determining this number. Since hierarchical merges two clusters at every step, they measured the dissimilarities of all the merging clusters. In compete linkage, they corresponds to the maximum dissimilarity within the new formed cluster. If the dissimilarity between a merge is high, it means that two rather different cluster were joined, so it is better to stop the merging before such a jump takes place. To make

this decision the dissimilarities are represented in a graph, that has on the x-axis the number of clusters decreasing, and on the y-axis the distance of the cluster merged and the points of the graph are connected by segments. The likely point (or points) to be chosen is the one before a steep segment that connects the following point of the plot.

### 3.6.2 Automatic clustering

The technique proposed by Surdeanu, Turmo, and Ageno will be mentioned as automatic clustering. The reason for this method comes from the fact that nowadays data mining, with all the resources available, is becoming more difficult to be handled manually, and, even if unsupervised techniques do not require data to be labeled, it was shown that algorithms like hierarchical clustering and k-means require the choice of a parameter by a human. So they wanted to increase the automation of the clustering process in a procedure that elects an adequate number of clusters by itself, without human decision. In [4] the authors based their work on the studies of Caliński and Harabasz [28], that proposed maximizing the ratio of between and within cluster distances as a method to detect the number of clusters.

Surdeanu et al.'s method searches for the best model in all the clusters created from the hierarchical clustering:

1. the dendrogram's clusters are sorted descending by their *quality* that is expressed by some quality measures that intuitively assess the likeliness that a cluster contains all and only similar elements. The higher it is, the best the cluster is;

2. from the clusters ordered the first that contains a certain percentage of all the dataset is chosen. This percentage $\gamma$ represent the factor of confidence given to the hierarchical clustering algorithm;

3. lastly the obtained clusters in the previous step are filtered to remove elements that are already contained in bigger clusters. The result of this step is a candidate with a $\gamma$ confidence and a certain quality measure, since there are many as it will be explained later.

The ratio defined by Caliński and Harabasz is the score of a model. Maximizing its value means finding the model that has well separated clusters that within have very similar elements. Indeed it is calculated as:

$$C = \frac{B(n-k)}{W(k-1)} \tag{3.9}$$

where $B$ is the between distance, i.e. the distance between each cluster and the others and $W$ is the within distance, that measure the distance of the elements of a cluster to their centroid.

$$\mathbf{B} = \sum_{i=1}^{k} n_i \, dist(centroid, meta\_centroid)^2 \tag{3.10}$$

$$\mathbf{W} = \sum_{i=1}^{k} \sum_{j=1}^{n_i} dist(d_j, centroid)^2 \tag{3.11}$$

$n$ is the total number of elements of the dataset, $k$ is the dimension of the current model, $n_i$ is the size of the $i$-th cluster, $centroid_i$ is the mean of the elements of a cluster and *meta_centroid* is the mean of the all dataset.

The algorithm of this procedure can be describe as follows:

**Algorithm 4:** Automatic clustering algorithm

```
bestModel = null, bestScore = 0;
forall quality measures do
    currentScore = first local maximum of C as γ decreases from 100% to 0% ;
    if currentScore > bestScore then
        bestModel = model associated to current quality measure and γ;
        bestScore = currentScore
    end
end
return bestModel;
```

#### 3.6.2.1 Quality measures

Surdeanu et al. used different quality measures to asses the quality of the clusters in the dendrogram, that starts from 4 observations.

**Minimising the within distance** corresponds in having clusters that contain objects that are similar, thus that are closer to each other.

$$\mathbf{W}(c_i) = \frac{1}{n_i(n_i - 1)} \sum_{x_r \in c_i} \sum_{x_s \in c_i, s \neq r} dist(x_r, x_s) \tag{3.12}$$

is the measure that aims to have small pairwise distances of the objects of the clusters. It favours clusters with small **W** values.

Objects that are similar should be contained in one cluster and be well separated from others clusters, so **the between distance of clusters should be maximised**. The measure **B** is the following:

$$\mathbf{B}(c_i) = \frac{1}{n_i(n - n_i)} \sum_{x_r \in c_i} \sum_{x_s \notin c_i} dist(x_r, x_s) \tag{3.13}$$

that calculates the pairwise distance of objects in the $i$-th cluster with the remaining objects of the dataset.

Using **B** and **W** as post-filtering function of clusters may cause issues: **B** tends to be large for most clusters, as the criteria for clustering algorithms is to maximise inter-cluster distance; **W** instead has great variations as it is applied through all the dendrogram, where clusters have very different sizes. So in clustering comparing functions, **W** has more influence. Thus **maximising the distance in the cluster vicinity** allows to measure the separation of a cluster with just its neighbours, without introducing the noise of the whole collection.

$$\mathbf{N}(c_i) = dist(c_i, sibling(c_i)) \tag{3.14}$$

Using **W** and **B** as quality measures has also two potential drawbacks: they favor small and compact clusters, separated from the rest of the dataset and groups represented by denser cluster. In the first case the system produces many categories with smaller coverage, in the other one it will miss information in the ignored categories. So it is necessary to pay attention also to other properties of clusters, apart from the density, like the growth **G**, characterized as the cluster expansion at the last merge occurred in the dendrogram, relative to the density of the cluster's two children. It is defined as the ratio of the between distance of the children $c_{i1}$ and $c_{i2}$ and the average of the pairwise distances between the objects within the two children.

$$w\_sum(c_i) = \sum_{x_r \in c_i} \sum_{x_s \in c_i, s \neq r} dist(x_r, x_s) \tag{3.15}$$

$$within\_children(c_i) = \frac{w\_sum(c_{i1}) + w\_sum(c_{i2})}{n_{i1}(n_{i1} - 1) + n_{i2}(n_{i2} - 1)} \tag{3.16}$$

$$\mathbf{G}(c_i) = \frac{dist(c_{i1,}, c_{i2})}{within\_children(c_i)} \tag{3.17}$$

where $w\_sum(c_i)$ is the sum of all distances between objects within the cluster $c_i$ and $within\_children(c_i)$ is the average distance between objects of the children of the cluster. Good models have a **small growth factor** that means two close clusters are merged; on the other hand a big growth factor means that two far and distant clusters were joined.

From this observations Surdeanu et al. [4] derived 6 quality measures, where observations that have to be maximized (**B** and **N**) are at the numerator of the formula, while the others that have to be minimized are at the denominator (**W** and **G**).

Table 3.2 Quality measures

| Name | $W$ | $WB$ | $WN$ | $GW$ | $GWB$ | $GWN$ |
|---|---|---|---|---|---|---|
| Formula | $1/\mathbf{W}$ | $\mathbf{B}/\mathbf{W}$ | $\mathbf{N}/\mathbf{W}$ | $1/\mathbf{GW}$ | $\mathbf{B}/\mathbf{GW}$ | $\mathbf{N}/\mathbf{GW}$ |

These formulas will be implemented in the part of the algorithm *Automatic clustering.*

## 3.7 Clustering comparison

Since this work will adopt two different techniques on the same data, it is important to know how they perform and how similar they are. For this reason is important to compare them, not only in a qualitative way, that could be subjective, but also with a quantitative technique that can asses numerically the similarity of the two methods, and thus the correctness of the automatic clustering.

A good source to find methods to quantify this correspondence comes from Wagner and Wagner [29]. They reviewed a series of measures to compare different clusterings, as the applications for this investigation are many, like checking if the algorithm is too sensitive to small perturbations or if order of the data can produce very different results or see how does a clustering compare to an optimal solution. Wagner and Wagner classified the measures in 3 groups:

1. counting pairs of elements;
2. summation of the set overlaps;
3. use of the mutual information.

In the first category the various measures count pairs of objects that are in the same cluster in both clusterings, so that are classified in the same way. Specifically it is

necessary to count the elements that are all in the same or all in different clusters for both clusterings or if they are in the same in one clustering but nor in the other, then calculating this for all the pairs gives the result of the similarity. This measures can also be calculated through the *confusion matrix*, i.e. a matrix where the $ij$th element is the number of elements in the intersection of two clusters of the two clusterings.

However this measures have drawbacks: some of these are sensitive to certain parameters like cluster size or number of clusters; others make strong null hypothesis assumptions like independence of the clusterings, although this is not the case of this work, but also ones that are more strict like fixed number of clusters and fixed cluster sizes, indeed none of the algorithms work with the last requirements.

The other group of measures match clusters that have the maximum overlap, absolute or relative, but they don't take into account the unmatched parts of the clusters. There are case where, given two clusterings variation of the original one, where in one of the two, part of the elements of one cluster is reassigned to the subsequent one, while in the other this part is reassigned evenly between all clusters, this kind of measures produce the same value, but it is obvious that the two clustering variations are not the identical.

Lastly the category of mutual information is based on the entropy applied to clusterings that informally represent the measure for the uncertainty about the cluster of a random chosen element. It can also applied to two different clustering and takes the name of *mutual information* that describes how much can we reduce the uncertainty about the cluster of a random element when knowing its cluster in another clustering.

The pros about this group based on information-theoretical considerations is that it does not have the drawbacks of the others groups, like strict requirements and assumptions. However these measures are quite recent and they are not deeply treated in the literature, so they might suffer from drawbacks that are not yet discovered.

Given that the first group is composed of measures that have too many limitations while the third one is still not well studied, the choice in this work was for the second group, in particular the *Maximum Match Measure* $\mathcal{MM}(\mathcal{C}, \mathcal{C}')$: it searches in the confusion matrix $M$ for the largest entry $m_{ab}$ and match the corresponding clusters $C_a$ and $C_b$, removes the $a$th row and $b$th and repeat this step until the matrix has size 0. All the entries are summed up and divided by the total number of elements.

$$\mathcal{MM}(\mathcal{C}, \mathcal{C}') = \frac{1}{n} \sum_{i=1}^{min\{k,l\}} m_{ii'} \tag{3.18}$$

where $i'$ is the index in of the cluster in $\mathcal{C}'$ clustering. If the number of clusters in the two clusterings is different $k \neq l$, this measure does not take into account the $|k - l|$ remaining clusters in the clustering with bigger cardinality.

## 3.8 Classification

With classification it is meant a technique to find a the right group or class to new elements given a previous knowledge. The previous knowledge can be a training set, a set of observation to classify to find their corresponding clusters. In this work classification was implemented using a previous clustering, that differs from supervised classification, indeed the data used here is not labelled, but the classification obtained is tested against the results of a normal clustering. This technique was adopted to verify how the algorithm behaves with a variation in the dataset and to see if the results are consistent.

# Chapter 4

# The implementation

In order to realise this project, the platform chosen was *Python*. Python is a programming language born in the late 1980s that through the years has gain a lot of success and has been updated many times, reaching the actual version 3.7.0. It is an interpreted language, which means that it does not need a compiler, and the interpreter is in charge of the task of portability, so there is no need to adapt the code to different platforms, as the output from the interpreter will always be the same. It has a clean syntax and is dynamically type, which means that variables type is never declared. It follows Object Oriented Programming and allows also the paradigm of functional programming. The programmer can count on many libraries, both built-in and portable, and furthermore it allows cross-application communication and can load C/C++ libraries [30].

Python has now many applications, in particular in the field of web development. It is largely used in backend programming of web applications. Along this also GUI programming and software prototyping has a discrete success.

But the reason for its choice is that it has become very common for scientific and numeric computing. Python's popularity in this field is possible thanks to the fact that it is a free and open source language, so programmers can modify the code by their needs. Then there are also many powerful libraries specific for this area of study, like NumPy, SciPy, Pandas, matplotlib and more, and some of these are used in this project. Thanks to these factors, Python became preferred over other paid solutions like Matlab.

## 4.1 First attempts

Initially the idea was to use one of the many libraries dedicated to machine learning available. *SciPy* is one these, indeed it has many functions for clustering and hierarchical

clustering was already realised and optimised, ready to be used, like *scikit-learn* another powerful library with an implementation of k-means. However testing SciPy library it was shortly clear that it was not suitable for this scope: it was easy to use, but too simplistic to be used with the data of this project. It allowed to use few implemented distance functions, Euclidean by default, but it was not possible to define a custom one that take into accounts u and v values, along with timestamps, i.e. the distance function 3.8.

So instead of using modules of Python it was decided to write manually the algorithm for the hierarchical clustering and k-means.

## 4.2 Data Loading

The first step of the solution presented is the data loading. When running the application, the user is prompted with a window where he can choose the values of the parameters.

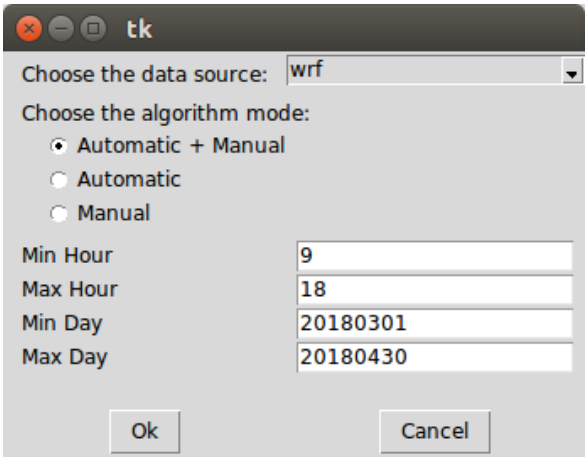



Fig. 4.1 Initial screen of the application.

First which of the two available data source, WRF or AROME. Secondly if he wants to test both the automatic and manual clustering or not. Then the parameters for the date and time. In figure 4.1 are displayed the default values. The interval 9-18 was chosen as the sailing race will take place during the day, so it was not meaningful to load the night files that could have influenced the results. The default values for the days take the whole period of data available.

As said in 2.3.1.1 the WRF files are csv and numpy library has a function to load them, *genfromtxt*. To represent a timestamp and its measurement was used a custom

class, *HourData*, characterised by a date and a list of 16 *Parameters* one for every coordinate (cf. 2.3.2), a different class with fields for the coordinates and the weather parameters. So, an HourData object contained all the locations parameters. It chosen to group all the locations in a single object because the data is more often accessed by the date and time rather than by coordinates. Thus, since different parameters of a timestamp measurement where split in different files, in the program all the 6 different files for each timestamp (one per each weather parameter, cf. 2.2) were merged, so, for every file read was necessary to verify if the object with the timestamp read was already created and in this case, updates its list of parameters.

On the other hand, with AROME a single file contained all information of the locations, so the object for a given timestamp was built in a single step, without searching the dataset for existing objects, like in WRF. With this kind of files, the reading was made with netCDF4 library, accessing the parameters, or Bands, inside the files and reading the values of the 16 locations.

**Algorithm 5:** Data loading procedure

**input** : data source, algorithm mode, time and date intervals
**output** : dataset

Shows the application window with inputs;
Read inputs within date and hour ranges;
**if** *data source is WRF* **then**
    read files;
    **if** *timestamp already in dataset* **then**
        retrieve the correspondent HourData object and update it;
    **end**
    **else**
        create a new HourData object and add it to the dataset;
    **end**
**end**
**else if** *data source is AROME* **then**
    read files one by one, create an HourData object and add it to the dataset
**end**
**return** the dataset;

After these steps, the dataset is loaded and ready to be used.

## 4.3 Normalisation

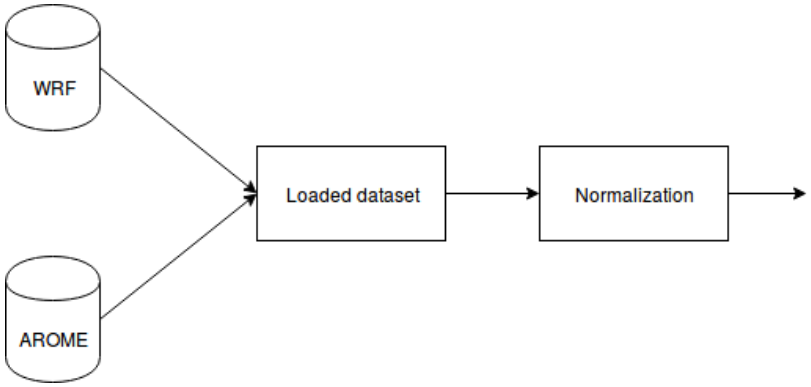



Fig. 4.2 Section of data normalisation.

Before diving into the calculation, the data must go through a preliminary process, called *normalisation*. The normalisation is a preprocessing step, necessary when a distance function is used, because it is required that all the objects have the same weight, so that the distance is not influenced by a scale factor of the data [31, 32].

There are different types of normalisation but this process depends on the data used. In this scope the normalisation used is the one defined by Kaufmann and Whiteman [2] that intended to prevent situations where stations characterised by high wind speeds could overweight other stations and measurements in the distance function. The normalisation adopted normalises $u$ and $v$ component first by the time-average speed:

$$u'_{ij} = \frac{u_{ij}}{s_j}, \qquad v'_{ij} = \frac{v_{ij}}{s_j}, \tag{4.1}$$

that means $u$ and $v$ at each time $i$ and station $j$ were normalised by the time-average speed $s_j$ at each site, that is:

$$s_j = \frac{1}{M_j} \sum_{i=1}^{M_j} \sqrt{u_{ij}^2 + v_{ij}^2}, \tag{4.2}$$

where $M_j$ is the total number of hourly winds at the site $j$. After this normalisation of the hourly wind measurements, the individual wind patterns were normalised in the following way:

$$\tilde{u}_{ij} = \frac{u'_{ij}}{s'_i}, \qquad \tilde{v}_{ij} = \frac{v'_{ij}}{s'_i} \tag{4.3}$$

where $u'_{ij}$ and $v'_{ij}$ are the normalized values calculated previously, and $s'_i$ is the spatial-average speed at each time $i$:

$$s'_i = \frac{1}{N_i} \sum_{j=1}^{N_i} \sqrt{{u'_{ij}}^2 + {v'_{ij}}^2}, \tag{4.4}$$

and $N_i$ is the total number of sites at time $i$.

Thanks to the normalisations 4.1 and 4.3, wind flows that differs by a scaling factor are grouped together.

In the application a copy of the dataset is normalised, since the original values of the data are necessary for later calculations and for the report.

## 4.4 Hierarchical Clustering

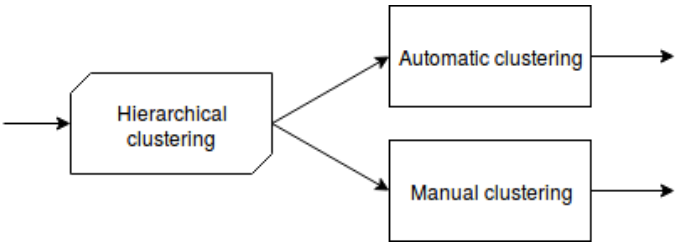



Fig. 4.3 Section of hierarchical clustering.

After the normalization it is the turn of the core of the program, the hierarchical clustering. As explained in section 3.6 two variations of hierarchical clustering are run, the automatic and the manual.

### 4.4.1 Automatic clustering

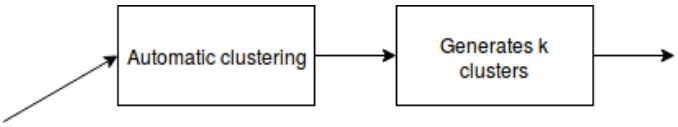



Fig. 4.4 Section of automatic hierarchical clustering.

The algorithm implemented follow closely the one described in section 3.6.2. Firstly the distance matrix is built to track the distances of every cluster, which at the beginning will contain only one object. Then is the turn of the dendrogram, that is created

thanks to a custom class that extends a library made to build trees, called *anytree*. At the creation the dendrogram contains only the leaves with one object.

The clustering process progresses searching for the two closest clusters in the distance matrix, it merges them and then the distance matrix is updated with the newly formed cluster, while in the dendrogram the new cluster is added as parents of the respective leaves. This procedure continues until there is only one cluster left that contains all the elements of the dataset.

After that the dendrogram is built, the next step is the quality measures calculation. Every node of the dendrogram is processed to calculate each of the six quality measures indicated in table 3.2. In order to optimise the calculations, the quality measures are computed only if the node has more than one object, since the $W$ observation formula 3.12 has the denominator $n_i(n_i - 1)$, this would return an error, and neither would be possible to measure the distance to other objects. For this reason the quality measure $W$ in this case was set to 0, and since all other quality measure depends on $W$, they were not calculated but saved as zero. The same method was applied for quality measures that depended on $\mathbf{G}$ as a leaf does not have any child.

Once all the quality measures are calculated and their results saved in as many arrays, the score calculation algorithm looks for the best model. The result of this process are the k clusters, but they are not complete yet, as, based on the confidence $\gamma$, not all the objects are contained in those clusters, so the excluded objects are assigned to the closest cluster. After this reassignment, the clusters are formed and ready to be processed by k-means.

**Algorithm 6:** Automatic hierarchical clustering

```
input  : dataset
output : k clusters
Initialize the distance matrix;
initialize the dendrogram;
while the number of cluster ≠ 1 do
    look for the two closest clusters;
    merge them;
    update the distance matrix and the dendrogram with the new cluster;
end
foreach node of the dendrogram do
    calculate the quality measures;
end
sort the quality measures arrays descending;
bestModel = null, bestScore = 0;
foreach quality measure do
    γ = 100%;
    while γ ≥ 0 do
        load the γn best nodes of the current measure;
        calculate the score C;
        if current score > bestScore and local optimum then
            bestModel = current model;
            bestScore = current score;
        end
        γ = γ - 10
    end
end
reassign elements out of γ return the clusters;
```

### 4.4.2 Manual clustering

The clustering core of the so-called manual clustering does not differ much from the automatic. It is somewhat simpler, since it does not calculates the distance measure.

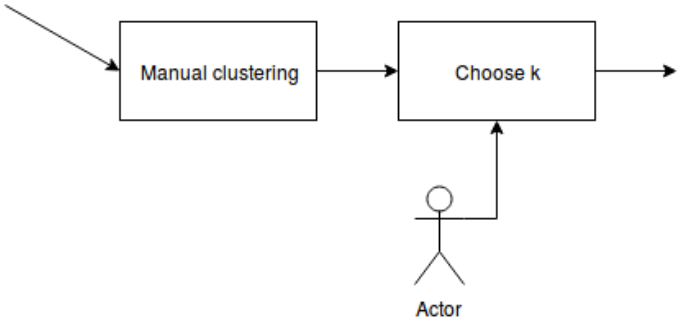



Fig. 4.5 Section of manual hierarchical clustering.

Similarly to the automatic, the first step is the creation of the distance matrix, however the dendrogram is built differently: since not all the nodes of the dendrogram are necessary, as it is very unlikely that the chosen number of clusters is a high value, in order to save memory only the current active clusters are kept saved in an array.

Another difference with the automatic clustering are the dissimilarities that are calculated at every merge, to make a graph and help decide the correct number of clusters.

**Algorithm 7:** Manual hierarchical clustering

**input** : dataset
**output** : k clusters

Initialize the distance matrix;
add every object to an array;
**while** *the number of cluster* $\neq$ *1* **do**
    look for the two closest clusters;
    merge them;
    measure the dissimilarity;
    update the distance matrix and replace the old cluster with the new one;
**end**
show the dissimilarities graph;
rerun the algorithm but stops when the number of cluster == the number chosen;
**return** the clusters;

It is worth mentioning that the algorithm is run twice: the first is to merge all the object in one cluster and calculate the dissimilarities, the second to obtain the clusters formed. This was planned to save memory. Executing twice the algorithm did not affected much the performance, indeed the processing time was much faster compared

to the automatic clustering, that requires 5 minutes [1] for the whole process, while the manual clustering can complete the computing in around 50 seconds [1] using WRF dataset. This however can be object for further work.

## 4.5 K-means

The resulting clusters of both automatic and manual clustering are the input for k-means, but before proceeding with the algorithm it is required to define a threshold value, needed in the *Wishard's variant* (3.4.1).

The procedure to use an appropriate threshold comes from Kaufmann and Weber [3], where for each cluster, the distance of its mean to other clusters is calculated, and then the frequency distribution was analysed, choosing a local minimum. This step was preliminary to k-means execution.

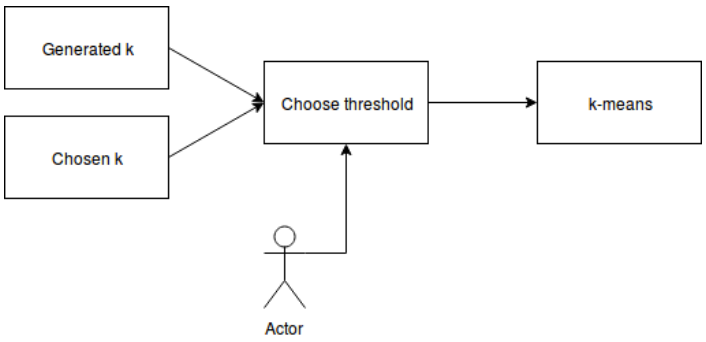



Fig. 4.6 Section of k-means.

So firstly the user pick a threshold, then starts the k-means execution.

[1] Using an Intel i7 quad-core processor, 16 GB of RAM computer

**Algorithm 8:** Threshold choice procedure

```
input  : clusters from hierarchical clustering
output : threshold
foreach cluster do
    calculate centroid;
    foreach other objects of the dataset do
        calculate distance to centroid;
        collect information on the distribution;
    end
end
plot the distance distribution;
read choice of the user;
return the threshold;
```

**Algorithm 9:** K-means procedure

```
input  : clusters from hierarchical clustering
output : final clusters
repeat
    foreach cluster do
        calculate centroid;
    end
    foreach objects of the dataset do
        calculate distance to all centroids;
        if distance to closest centroid > threshold then
            put element in outliers;
        end
        else
            put element in the cluster of the closest centroid if different to its
             cluster;
        end
    end
until there are no more changes;
if any cluster has size == 1 then
    put element of this cluster in outliers;
end
return the clusters;
```

So the program asks for an appropriate threshold and it runs the k-means algorithm that allows objects to change cluster and move to the most appropriate one, operation that is not possible with hierarchical clustering. The threshold could be chosen automatically picking the local minimum, but currently for testing purposes it was preferred a choice by the user. The result of the algorithm is the final clustering.

## 4.6 Results report

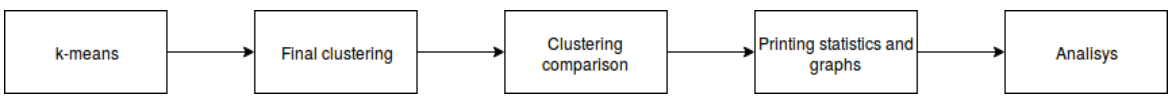



Fig. 4.7 Section of results report and analysis.

The end of the execution of k-means indicates the end of the clustering algorithm and the final composition of clusters is available. At this point the clusters are analysed to extract the characteristics that distinguish each cluster and to compare automatic and hierarchical clusterings. Here a report file is created that contains statistics and information about the clusters, useful for the meteorologist to evaluate the grouping.

The clustering comparison in 4.7 represents what is explained in section 3.7, a quantitative measure to assess similarity between clusterings. Other than that, the common elements of the clusters were studied, to see if grouping contained similar elements. It was implemented in the following way.

**Algorithm 10:** Clustering comparison procedure

**input :** hierarchical and automatic clusterings
**output :** Maximum Match Measure

Build the confusion matrix;
**repeat**
  take the highest value in the confusion matrix;
  count the identical elements in the matching clusters;
  update the Maximum Match Measure;
  remove from the confusion matrix the row and column corresponding to the clusters;
**until** *there are no more elements in confusion matrix*;
**return** the Maximum Match Measure;

Thus, the result of this procedure is the Maximum Match Measure, while the number of identical objects between clusters was saved and displayed in a table of the report file. The last comparison in particular is not a standard procedure, but an easy way to compare the similarity. Information on all of the clusters are included with this measure. First of all there are the logs coming from the execution, in particular the input chosen by the user, that are the days interval and the hour range. In addition the execution time of the algorithms is included to evaluate the performance, as well as information on the results, like the number of clusters resulting from the hierarchical clustering and the updated number after k-means execution. Plots of the threshold that are showed to the user are included too.

After the logs a series of tables contain information on the composition of clusters. The first characteristic is the number of elements, then the maximum and minimum values of each parameter and their averages. Subsequently there are the ranges of the

parameters: the minimum and maximum values of the complete dataset are taken and the interval is divided into 5 ranges and for each cluster it is showed the percentage of the distribution. These results in particular were really useful for the meteorologist to check the clustering, especially wind direction and speed. Ideally a cluster should have winds flowing in a certain direction with a certain speed and be different from other clusters. These values are calculated examining each element of the clusters, keeping track of their means and ranges.

More information is provided by the transition matrix. It shows how the clusters change to one another. To do so, it was checked to which cluster belong the following timestamp of a given element of the cluster, representing in percentages these values. So the representation is a square matrix with as many rows and columns as the number of clusters. The intersection of a row $a$ and a column $b$ is the percentage of elements that transition from cluster $a$ to cluster $b$. For example if 5 out 10 elements have the following element in the same cluster, the matrix will contain 50% on the diagonal, while on the other columns the rest of the 5 elements that transition to the other clusters.

Lastly the hourly distribution of the winds is expressed with a bar plot for each cluster. Dividing the abscissa axis for each hour in the time span 9 to 18, the ordinate axis is the percentage of elements of the cluster that occurs at that hour. This graph is an immediate visualisation of the distribution of the winds to help the meteorologist in the analysis.

All these information are returned in the form of a pdf file to be analysed by a meteorologist to check the validity of the clustering.

## 4.7 Classification

Along with the unsupervised clustering, it was implemented a classification part to test the validity of the implemented clustering. It was planned to use the same WRF dataset, except for a group of days, and run the automatic hierarchical clustering. The result of the execution was then compared with the results of the whole dataset. The days were chosen spread in the period of the available data, i.e. every $8^{\text{th}}$, $17^{\text{th}}$ and $26^{\text{th}}$ day of each month, and this included the whole days, so from 9 to 18.

The implementation was straightforward: firstly is loaded the dataset without the mentioned days and the automatic clustering is executed. Then the automatic clustering is run with all the days of the dataset and the results of the two are compared. To do this the clustering comparison illustrated in the previous section, and also a

pairwise comparison of the clusters that contained each timestamp to check if the elements contained are similar. Again, this is not a standard procedure, but a useful way to see if the elements classified were grouped with the same objects.

**Algorithm 11:** Classification procedure

Load the dataset without the chosen days;
run the automatic algorithm;
Load the original dataset;
run the automatic algorithm;
Run the clustering comparison with the results of the two automatic clustering;
**foreach** *element to be classified* **do**
    take the cluster of the first clustering that contains the element;
    take the cluster of the second clustering that contains the element;
    count and save the identical elements of the two clusters;
**end**
**return** *the number of identical elements of each pair of clusters for every object*

The results of this algorithm is saved in a table where each entry contains the timestamp, the clusters it belongs to and the number of identical objects in the clusters. This tables helps to evaluate the results of the classification and the correctness of the algorithm. Indeed, if the two clusters coming from the two different clusterings contain many elements in common, it means that the elements with certain values are grouped together, independently of the clustering type.

# Chapter 5

# Results Analysis

The program was tested with WRF dataset but it was not possible to compare the results with AROME data because the available data for the latter differ, as not all March and April are available, but only the period from the $25^{\text{th}}$ of March to the $16^{\text{th}}$ of May.

The result report generated at the end of the program execution, as illustrated in the previous chapter, is the source for the analysis. The statistics and information contained are helpful for the meteorologist to evaluate the clustering obtained, starting from the table containing ranges of winds directions and speed. These allows to understand how well the clustering performed, if the elements in the clusters are well separated by each other. The results obtained are illustrated in the Appendix.

## 5.1 Meteorological analysis of clustering results

Currently an analysis of meteorological data, able to lead to the identification of weather scenarios, having well defined weather characteristics, is performed manually by the user and for instance by the meteorologist.

A subjective interpretation of each day experienced on the field and the comparison with similar days experienced previously, allows for the identification of repeating weather conditions called *weather patterns*. In sailing, the most important weather variable is wind. However, it is important to stress that each wind pattern is strongly related to other meteorological variables, such as temperature, air pressure and humidity. Therefore, despite the main focus of the present work is the identification of repeating wind scenarios, we will refer to the more general name of *weather patterns*.

The first step for the identification of weather patterns for sailing, is to split the whole wind direction range (0-360 degree) into several smaller ranges or called

*direction sectors*. Indeed, the wind is behaving in a similar way within each specific wind direction sector. Secondly, additional information concerning the wind speed, the air temperature or the air pressure are also analysed to obtain a more in-depth categorisation.

The goal of the present work is to assess challenges and opportunities of using automatic clustering for the identification of weather patterns for sailing. The identification of these patterns can be performed either for a well-known site, where qualitative weather conditions are familiar to the user, but scientific causes for the existence of repeating weather conditions are not well-known. On the other hand the same approach can be used for a relative new sailing venue, to make a first assessment of existing weather patterns. In this case, the first identification is followed by further analysis to validate and calibrate the patterns. Since the chosen area is already known, the idea is to evaluate how much the clustering process has been able to capture the existing wind patterns and what information can be extracted from them, according to the different meteorological features that have been described in the previous section. An extensive analysis is completely outside the scope of this thesis, therefore we have just tried to draw a sample of conclusions that simply show that the automatic methodology used can make the meteorological analysis far easier. As this analysis requires additional meteorological knowledge, it has been carried out by the author of this thesis with the help of an expert in the field.

The usual wind direction ranges used to create weather patterns are the following:

1. NNE 0-45
2. ENE 45-90
3. ESE 90-135
4. SSE 135-180
5. SSW 180-225
6. WSW 225-270
7. WNW 270-315
8. NNW 315-360

So the categories derived from the automatic and manual classifications have been divided according to the above mentioned direction sectors.

### 5.1.1 Automatic clustering

The following results come out from the analysis of the automatic classification table A.8 of the appendix:

- Cluster 1 and 2 belong to NNE wind sector, both with a percentage of data about 90%
- Cluster 3 and cluster 13 belong to both NNE and ENE sectors, with about 90% of the data. However cluster 3 has more data belonging to the ENE sector and the 13 has more data belonging to NNE sector
- Cluster 4 has about 90% of the data belonging to sectors ENE and ESE, with the majority of the data, 75% , belonging to sector ESE
- Cluster 5 has, similarly to cluster 4, 75% of the data belonging to sector ESE but the rest of the data belong to sector SSE. So this cluster represent data coming from a ″more right direction” compared with the cluster 4
- Cluster 6 has 90% of the data belonging to sectors NNW and NNE
- Cluster 7 is having about 90% of the data belonging to sector ENE. So this cluster might have some similarities with cluster 3 and 13
- Clusters 8 and 11 have about 90% of the data belonging to sector WSW
- Clusters 9 and 10 have about 70% of the elements inside sector WNW
- Cluster 12 is the only one having more than 80% of the inside sector NW
- Cluster 14 represent mainly sectors SSE and SSW, so mainly southerly winds
- Cluster 15 represents sector WSW with 85% of the elements, while cluster 16 has 36% of the elements within the same sector of 15, so WSW, but 45% of the elements within sector WNW
- Finally cluster 17 is having elements not belonging to one specific sector. Data are going from sector NNE to E

An analysis of the above mentioned results, shows that different clusters contain elements having very similar directions. It is therefore difficult to understand the reason why different clusters exist. Therefore, an analysis of other meteorological

characteristics such as wind speed, atmospheric pressure and temperature has been performed to identify specificities for each cluster.

Probably due to the little amount of data, it is hard to find reasons why data belong to different clusters. However, for some of the clusters some interesting information can be found.

For instance, cluster 8 is one of the clusters having the higher values of atmospheric pressure and the lowest values of wind speed. It is also one of the clusters having no precipitation at all. Finally, it has temperatures going up to 22° C, so it is one of the warmer clusters.

On the other hand cluster 11, which is similar to cluster 8 in wind direction and speed, is for sure a cluster characterised by colder temperatures, having only 1% of the data going above 16° C. It also has values of atmospheric pressure lower than cluster 8.

Patterns include an additional useful information, which is the probability of transition from one pattern to another along the day. This allows to predict the behaviour of the wind whenever we identify that one set of data belongs to one cluster (or wind pattern). In our case, it is also interesting to notice that the transition from cluster 8 to other clusters is in the most of the cases, 44% , to cluster 11. Therefore, it is likely that cluster 11 represents an initial phase of a WSW gradient wind, colder and with lower values of pressure. Once the heating start to affect the atmospheric conditions, there is a transition to cluster 8.

Another interesting information is coming out from the analysis of cluster 6. Actually we noticed that in the most cases the transition is to cluster number 2. On the other hand, as previously mentioned, just by looking at wind direction clusters 1 and 2 look pretty similar. A more in-depth analysis shows that cluster 1 almost only goes to cluster 2 and vice versa. Therefore cluster 2 should have some specificities relating it to cluster 6. Cluster 2 has a bit lower values of the speed compared with cluster 1, similarly to cluster 6. Moreover cluster 6 seems to be mainly related with morning hours. Therefore a possible scenario can be a morning light NNW-NNE, going to light NNE (cluster 2) and finally going to a bit stronger NNE (cluster 1). In case of dropping of the speed, the cluster 1 will change to cluster 2. In addition, cluster 2 has a bit more probability of being associated with some precipitation, while cluster 1 not. This point justifies even more the lighter speed related with cluster 2.

Finally, the analysis of the transition from cluster 15 to other clusters shows that most of the times cluster 15 changes to cluster 9, and only in few cases to cluster 10. However, by the analysis of the wind direction, cluster 9 and 10 are very similar. Both are characterised by light wind speed values, being cluster 10 just a bit lighter than

9. Cluster 10 has a bit higher values of atmospheric pressure. However, no significant differences in Temperature, Precipitation of Humidity parameters are found.

One of the main issues of the above mentioned classification is that some data are split into different clusters, despite having very similar values of wind speed, direction and of additional meteorological variables. In these cases, it is therefore very difficult to understand exactly, what each cluster represents and why it changes to another.

### 5.1.2 Manual clustering

The same analysis has been performed for the two manual classifications described in the previous section, the one containing identified as ″Manual 10" (though it contains 9 clusters when the outliers are eliminated) and the one identified as "Manual 17" (containing 14 clusters when outliers are eliminated). As we did with the automatic clustering, we are going to ignore the cluster of outliers, since we consider that it does not provide any significant meteorological information.

The table below shows which clusters belong to the different direction sectors for all the three classifications.

Table 5.1 Clusters division by direction

| | Automatic | Manual 10 | Manual 17 |
|---|---|---|---|
| 0-45 | 1, 2, 3, 6, 13, 17 | 1, 3, 9 | 1, 2, 4, 12, 14 |
| 45-90 | 3, 4, 7 | 5, 9 | 2, 5, 14 |
| 90-135 | 4, 5, 13 | 4, 9 | 5, 7, 14 |
| 135-180 | 5, 14 | 7, 9 | 10, 14 |
| 180-225 | 14 | 7 | 11 |
| 225-270 | 8, 9, 11, 15, 16 | 2, 8 | 3, 8, 9, 13 |
| 270-315 | 9, 10, 16 | 6 | 6, 8, 9 |
| 315-360 | 6, 10, 12, 17 | 3, 6 | 4, 6, 12 |

Firstly, it should be noticed that cluster 9 of the Manual 10 and cluster 14 of Manual 17, are quite spread all over the direction sectors and might be considered as additional outliers.

Secondly, without considering cluster 9, the Manual 10, due to the limited number of clusters, is the only one where every direction sector corresponds to one or maximum two clusters. This does not mean, however, that its division into actual patterns is more accurate than the clustering with more clusters, as commented below.

Both the automatic and the Manual 17 have mainly two or more clusters inside every direction range. Therefore, by comparing the Automatic cluster and the Manual 17 clusters, no particular advantage is noticed in performing a manual classification.

On the other hand, a more comprehensive analysis of the Manual 10 clustering has been performed, and in particular starting from the transition matrix.

Cluster 1, NNE wind, when changing to another cluster, goes to cluster 3 or 5. Cluster 3 contains both element belonging also to cluster 1 or elements of the NNW sector. Cluster 5 is representing ENE winds. Therefore it is important to understand in which cases there is the transition from 1 to 3 or from 1 to 5. Analysing the wind speed direction, it is evident that cluster 3 represents lighter wind than cluster 5. Moreover cluster 5 has higher wind speeds than cluster 1. Therefore one can conclude that increasing NNE wind will most likely change to ENE, while decreasing NE winds will most likely stay within the same sector or change to NNW. This partly confirms the analysis made on the transition from cluster 6 to clusters 1 or 2 within the automatic classification. A point which confirms the analysis made also by the automatic classification is that cluster 3 has more chance of precipitation than clusters 1 or 5. Therefore in case of precipitation the scenario of lighter winds changing from NNE to NNW is most likely.

In conclusion, by performing this last analysis on the Manual 10 classification, we can sum up the following advantages and disadvantages:

1. An automatic classification is faster and repeatable but generates many clusters, some of them containing very similar elements.

2. The analysis of the results derived from the automatic clustering requires a more in depth 'human interpretation' to identify main reasons and probability of the transition from one class to another.

3. A manual clustering having a similar number of clusters to that of the automatic clustering does not present any particular advantage.

4. A manual clustering with less clusters than the automatic one, requires less human effort to find out conclusions. On the other hand, conclusions that can be derived both by the Manual 10 and the automatic clustering are very similar

### 5.1.3 Automatic clustering revisited

On the base of the above mentioned conclusions and in order to try to obtain well defined direction categories, it was decided to change table 5.1 in the previous section,

and analyse the results of only the automatic classification by using the following wind direction ranges (table 6a of the previous section):

| Min | Max | Direction sector |
|---|---|---|
| 0 | 22.5 | N |
| 22.5 | 45 | NNE |
| 45 | 67.5 | NE |
| 67.5 | 90 | ENE |
| 90 | 112.5 | E |
| 112.5 | 135 | ESE |
| 135 | 157.5 | SE |
| 157.5 | 180 | SSE |
| 180 | 202.5 | S |
| 202.5 | 225 | SSW |
| 225 | 247.5 | SW |
| 247.5 | 270 | WSW |
| 270 | 292.5 | W |
| 292.5 | 315 | WNW |
| 315 | 337.5 | NW |
| 337.5 | 360 | NNW |

We noticed that, with this new classification, the number of clusters having superposed wind direction ranges decreases significantly. This makes much easier the identification of unique wind direction sectors and the analysis of the probability of transition from one cluster to another.

A very interesting feature we can notice comes out from the analysis of the transition from cluster 16 to other clusters. Cluster 16 has mainly data representing mighty winds from SW to NW sectors. In 50% of the cases the transition is to cluster 10 (WNW), while in 50% of the cases is to sector 8 (SW) or 9 (W), with same probability. Cluster 8 is clearly the one related with lower wind speed, higher values of the pressure and lower probability of rain. While clusters 9 and 10 look very similar. One useful conclusion can be that, in case of initial light wind from SW to NW, dry weather, increasing pressure and no increase of the speed, the wind will go most likely to the SW. This theory is even more justified by the fact that cluster 16 is most likely in the morning while cluster 8 is most likely in the afternoon.

Analysis of transition between cluster 1, 2 and 3 is also significant. Clusters 1 and 2 have data belonging mainly to every hour of the day, while cluster 3 is clearly related with afternoon hours. Cluster 1 (N) can change to cluster 2 (NNE) or 3 (NE). On the other hand, cluster 2 mainly changes to cluster 1, so going from NNE to N, but not to cluster 3. Finally, cluster 3 changes to cluster 2 but almost never to cluster 1. Therefore the first conclusion can be that N wind can become NNE or NE. On the other hand, a NNE wind will mainly change to N and in very few cases to NE. By looking at the wind speed, we can notice that cluster 2 is lighter than 1 and 3. Therefore, the second conclusion is that, when increasing, the NNE wind will most likely become N. On the other hand, a N wind decreasing will become NNE, and if maintaining the same wind speed, it will become NE.

The latter are just examples of the information that can be extracted from the features of the identified clusters. We believe that this new analysis can be very helpful for two main reasons:

- It allows the identification of almost unique wind direction categories.
- It allows the identification of additional weather variables contributing to the transition from one cluster to another.

We can therefore conclude that, from a subjective meteorological point of view, the use of the automatic clustering offers slightly better results than the manual one, and the finer description of each cluster according to frequency of elements in 16 wind sectors makes far easier the identification of the weather patterns. Of course, it must be considered that we have only dealt with two months of data and therefore more thorough tests would be in order, but it allows us to confirm that it is a promising approach.

## 5.2 Classification results

Given the results obtained, in order to test our classification algorithm, it was necessary to check that the classification of the days chosen to be classified grouped them with the same objects as the normal clustering. Since there is no gold standard classification for the 2 months that are used as a dataset, in order to evaluate it there are two options:

1. Evaluate it manually, looking at the meteorological features of the day and looking at what cluster suits the best. This would be a subjective method dependent on the criterion of the meteorologist.

2. Computing the equivalence between the clustering with these days and the clustering without them. Then, it is possible to look if the clusters in which these days were originally in the complete clustering are equivalent to the ones in which they have been classified.

It was decided to adopt the second possibility, in order to have quantitative measures to assess this equivalence. It was taken the automatic clustering obtained in the previous section as a reference complete clustering. The equivalence between clusters is shown in table 5.2, and the results obtained for each days' timestamps are shown in table 5.4. The obtained Maximum Match Measure of the two clusterings is quite high and has a value of 83%. It can also be noted that, for 83% of the elements, the cluster in which the element has been classified is equivalent to the cluster in which the element was originally. It is reasonable to consider that these results confirm that the classification algorithm performs correctly.

Table 5.2 Clusters matching

| Cluster automatic with classification | Cluster automatic | Common elements |
|---|---|---|
| 1 (91) | 1 (87) | 84 |
| 3 (68) | 2 (68) | 61 |
| 4 (62) | 3 (60) | 57 |
| 5 (55) | 4 (55) | 54 |
| 8 (41) | 5 (37) | 37 |
| 2 (74) | 8 (34) | 32 |
| 9 (37) | 6 (37) | 30 |
| 10 (35) | 7 (35) | 29 |
| 6 (46) | 9 (30) | 27 |
| 7 (40) | 10 (28) | 23 |
| 12 (16) | 14 (19) | 14 |
| 11 (18) | 13 (21) | 13 |
| 13 (4) | 11 (26) | 0 |
| 14 (4) | 12 (25) | 0 |

Table 5.4 Clusters similarity

| Timestamp | Clustering w/ classification | Clustering w/o classification | Common El |
|---|---|---|---|
| 08/03 09:00 | 8 (41) | 14 (19) | 1 |
| 08/03 10:00 | 8 (41) | outlier | 0 |
| 08/03 11:00 | 8 (41) | outlier | 0 |
| 08/03 12:00 | 3 (68) | 17 (3) | 1 |
| 08/03 13:00 | 9 (37) | 6 (37) | 30 |
| 08/03 14:00 | 9 (37) | 6 (37) | 30 |
| 08/03 15:00 | 1 (91) | 1 (87) | 84 |
| 08/03 16:00 | 4 (62) | 3 (60) | 57 |
| 08/03 17:00 | 4 (62) | 3 (60) | 57 |
| 08/03 18:00 | 4 (62) | 3 (60) | 57 |
| 17/03 09:00 | 5 (55) | 4 (55) | 54 |
| 17/03 10:00 | 5 (55) | 4 (55) | 54 |
| 17/03 11:00 | 5 (55) | 4 (55) | 54 |
| 17/03 12:00 | 5 (55) | 4 (55) | 54 |
| 17/03 13:00 | 10 (35) | 13 (21) | 6 |
| 17/03 14:00 | 4 (62) | 3 (60) | 57 |
| 17/03 15:00 | 4 (62) | 3 (60) | 57 |
| 17/03 16:00 | 5 (55) | 7 (35) | 1 |
| 17/03 17:00 | 5 (55) | 4 (55) | 54 |
| 17/03 18:00 | 10 (35) | 7 (35) | 29 |
| 26/03 09:00 | 8 (41) | 5 (37) | 37 |
| 26/03 10:00 | 12 (16) | 14 (19) | 14 |
| 26/03 11:00 | 2 (74) | 15 (16) | 13 |
| 26/03 12:00 | 6 (46) | 9 (30) | 27 |
| 26/03 13:00 | 7 (40) | 10 (28) | 23 |
| 26/03 14:00 | 7 (40) | 10 (28) | 23 |
| 26/03 15:00 | 14 (4) | 6 (37) | 1 |
| 26/03 16:00 | 9 (37) | 6 (37) | 30 |
| 26/03 17:00 | 3 (68) | 2 (68) | 61 |
| 26/03 18:00 | 1 (91) | 3 (60) | 1 |
| 08/04 09:00 | 2 (74) | 11 (26) | 25 |
| 08/04 10:00 | 2 (74) | 15 (16) | 13 |
| 08/04 11:00 | 6 (46) | 15 (16) | 1 |
| 08/04 12:00 | 6 (46) | outlier | 0 |

Continues on Next Page...

Table 5.4 Clusters similarity

| Timestamp | Clustering w/ classification | Clustering w/o classification | Common El |
|---|---|---|---|
| 08/04 13:00 | 8 (41) | 5 (37) | 37 |
| 08/04 14:00 | 8 (41) | 5 (37) | 37 |
| 08/04 15:00 | 8 (41) | 5 (37) | 37 |
| 08/04 16:00 | 5 (55) | 4 (55) | 54 |
| 08/04 17:00 | 4 (62) | 3 (60) | 57 |
| 08/04 18:00 | 4 (62) | 3 (60) | 57 |
| 17/04 09:00 | 5 (55) | 4 (55) | 54 |
| 17/04 10:00 | 5 (55) | 4 (55) | 54 |
| 17/04 11:00 | 11 (18) | 13 (21) | 13 |
| 17/04 12:00 | 1 (91) | 1 (87) | 84 |
| 17/04 13:00 | 1 (91) | 1 (87) | 84 |
| 17/04 14:00 | 1 (91) | 1 (87) | 84 |
| 17/04 15:00 | 1 (91) | 1 (87) | 84 |
| 17/04 16:00 | 1 (91) | 1 (87) | 84 |
| 17/04 17:00 | 1 (91) | 1 (87) | 84 |
| 17/04 18:00 | 1 (91) | 1 (87) | 84 |
| 26/04 09:00 | 6 (46) | 9 (30) | 27 |
| 26/04 10:00 | 7 (40) | 12 (25) | 17 |
| 26/04 11:00 | 9 (37) | 6 (37) | 30 |
| 26/04 12:00 | 3 (68) | 2 (68) | 61 |
| 26/04 13:00 | 1 (91) | 1 (87) | 84 |
| 26/04 14:00 | 1 (91) | 1 (87) | 84 |
| 26/04 15:00 | 4 (62) | 3 (60) | 57 |
| 26/04 16:00 | 4 (62) | 3 (60) | 57 |
| 26/04 17:00 | 4 (62) | 3 (60) | 57 |
| 26/04 18:00 | 10 (35) | 7 (35) | 29 |

End.

# Chapter 6

# Conclusion

A subjective interpretation of weather conditions experienced each day on the field and the comparison with similar days experienced previously, allows for the identification of repeating weather conditions usually called ″weather patterns”. In sailing, the most important weather variable is wind. However, it is important to stress that each wind pattern is strongly related to other meteorological variables, such as temperature, air pressure and humidity. Currently an analysis of meteorological data, able to lead to the identification of wind patterns, having well defined characteristics, is performed manually by the user (normally a meteorologist). However, as happens in many other domains, nowadays we have an increasing number of available data, which makes very difficult for a human to analyse them all. The idea was to investigate if an added value to the human interpretation is derived from an automatic analysis of the data.

The specific goal of the present work has been to assess the challenges and opportunities of using automatic clustering for the identification of weather patterns for sailing. The identification of these patterns can be applied either for a well-known site, where qualitative weather conditions are familiar to the user, but scientific causes for the existence of repeating weather conditions are not well-known. On the other hand the same approach can be used for a relative new sailing venue, to make a first assessment of exiting weather patterns. In this case, the first identification is followed by further analysis to validate and calibrate the patterns.

We have initially based our methodology on existing work, but have extended it to make the system completely automatic, so that the intervention of the user will be restricted to the mere analysis of the thorough statistical results provided by the system. We have compared the performances of both the manual and automatic approaches, applied to numerical weather prediction models outputs. Due to computational constraints of producing numerical weather prediction data for long time periods, we have been able to test our system on a limited

amount of data: two months. Despite the limited dataset, very promising information was derived. For instance we can conclude that:

- Both the manual and automatic clustering systems provide very interesting information, as to the identification of almost unique wind direction categories (the wind patterns) and as to the identification of additional weather variables contributing to the transition from one pattern to another along the day. This allows furthermore to predict the behaviour of the wind whenever we identify that one set of data belongs to one cluster (or wind pattern).
- A manual clustering having a similar number of clusters to that of the automatic clustering does not present any particular advantage. In fact, for our limited test set and from a subjective meteorological point of view, the use of the automatic clustering offers slightly better results than the manual one, and the finer description of each cluster according to frequency of elements in 16 wind sectors makes far easier the identification of the weather patterns. Of course, it must be considered that we have only dealt with two months of data and therefore more thorough tests would be in order, but it allows us to confirm that it is a promising approach.

## 6.1 Future Work

Considering the conclusions reported above, a first step forward should be the test of the same approach on bigger volumes of data. In fact, an ongoing collaboration with the Meteorological Service of Catalunya (Meteocat) can allow the availability of longer time-series for the WRF model. This step would allow the assessment of the results obtained by the system with more data of the same type (in this case, spring weather patterns, which can be different from other periods of the year).

The second step should be, once an enough quantity of actual data will be collected in Tokyo 2020 sailing venue, to start testing the system with these data. Since we never worked with real data before, it might be useful to compute two clusterings, one with the output from weather models for the area (either Arome or WRF) and another one with the collected data for the same days, and evaluate that the results, though maybe not equal, are relatively consistent.

Thirdly, in our system, only wind data are used to infer the underlying clusters. Therefore, a possible extension might be the use of additional weather data (which would mean an obvious complication of the distance function).

Moreover the implemented code has room for improvements. Although the current performance of the program are quite good, the availability of more data would consequently means also more computational power needed, therefore code optimisation is required.

Obviously a simple laptop used for the development of this work would not be enough and the shift to server will be necessary, but, despite having a more powerful machine, some optimisations applied to code would grant faster results. Some of these could concern the manual hierarchical clustering that, as said in section 4, is run twice. Using a powerful machine allows to have more memory available, and keeping a complete dendrogram saved would not be a problem and, most importantly, would reduce the execution time. Another improvements that would benefit the time required to run the program could be the parallelisation of the task that regards the quality measures of the automatic clustering. It is noted as the most heavy process of the program and improvements would reduce the time to analyse big amounts of data.

Finally, there is a whole span of alternatives to extract additional information from the current data. It is clear that with the availability of more and more data, it is possible extract more specific information (according to the time of the year, to the different specific locations inside the area of study, to the dynamic evolution of the parameters... ). The idea would be to keep on extending the actual system by making it able to extract more and more useful information. We strongly believe that the real added value of data is not represented by only having a big amount of data but by making sense out of them, and this is particularly true while analysing complex and fast changing environmental systems.

# References


[1] M. Aran, J. C. Peña, and J. Amaro. Atmospheric circulation patterns associated with strong wind events in Catalonia, September 2009.

[2] P. Kaufmann and C. D. Whiteman. Cluster-analysis classification of wintertime wind patterns in the grand canyon region. *Journal of Applied Meteorology*, 1999.

[3] P. Kaufmann and R. O. Weber. Classification of mesoscale wind fields in the mistral field experiment. *Journal of Applied Meteorology*, 1996.

[4] Mihai Surdeanu, Jordi Turmo, and Alicia Ageno. A hybrid unsupervised approach for document clustering. *KDD*, 2005.

[5] THE NERC MST RADAR FACILITY AT ABERYSTWYTH. Wind vector notation conventions. URL http://mst.nerc.ac.uk/wind_vect_convs.html.

[6] National Center for Atmospheric Research. Weather research and forecasting model. URL https://www.mmm.ucar.edu/weather-research-and-forecasting-model.

[7] National Center for Environmental Information. Numerical weather prediction. URL https://www.ncdc.noaa.gov/data-access/model-data/model-datasets/numerical-weather-prediction.

[8] Jason Knievel. Numerical weather prediction (nwp) and the wrf model, 2006. URL https://ral.ucar.edu/projects/armyrange/references/forecastconf_06/02_wrf.pdf.

[9] National Centre for Meteorological Research. Arome. URL https://www.umr-cnrm.fr/spip.php?article120&lang=en.

[10] National Weather Service. Office note 388 grib1. URL http://www.nco.ncep.noaa.gov/pmb/docs/on388/.

[11] World Meteorological Organization. A guide to the code form fm 92-ix ext. grib edition 1. URL http://www.wmo.int/pages/prog/www/WMOCodes/Guides/GRIB/GRIB1-Contents.html.

[12] Unidata. Netcdf: Introduction and overview. URL https://www.unidata.ucar.edu/software/netcdf/docs/.

[13] Andrew Ng. Machine learning. URL https://www.coursera.org/learn/machine-learning/lecture/Ujm7v/what-is-machine-learning.

[14] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction to Statistical Learning*. Springer, 2013.

[15] Stanford Online Lagunita. Unsupervised learning. URL https://lagunita.stanford.edu/c4x/HumanitiesScience/StatLearning/asset/unsupervised.pdf.

[16] Justin Gage. Introduction to unsupervised learning. URL https://blog.algorithmia.com/introduction-to-unsupervised-learning/. Algorithmia.com.

[17] Vishal Maini. Machine learning for humans, part 3: Unsupervised learning, . URL https://medium.com/machine-learning-for-humans/unsupervised-learning-f45587588294.

[18] Vishal Maini. Machine learning for humans, part 4: Neural networks & deep learning, . URL https://medium.com/machine-learning-for-humans/neural-networks-deep-learning-cdad8aeae49b.

[19] Wikipedia contributors. Neural network — Wikipedia, the free encyclopedia, 2018. URL https://en.wikipedia.org/w/index.php?title=Neural_network&oldid=860697996.

[20] M Ester, H.P. Kriegel, J. Sander, and Xu. Dbscan: Density-based spatial clustering of applications with noise. URL http://www.cs.fsu.edu/~ackerman/CIS5930/notes/DBSCAN.pdf. Florida State University, Department of Computer Science.

[21] Aidong Zhang. Density-based approaches. URL http://www.cse.buffalo.edu/faculty/azhang/cse601/density-based.ppt. University at Buffalo, Department of Computer Science and Engineering.

[22] Wikipedia contributors. Expectation–maximization algorithm — Wikipedia, the free encyclopedia, 2018. URL https://en.wikipedia.org/w/index.php?title=Expectation%E2%80%93maximization_algorithm&oldid=860035011.

[23] Wikipedia contributors. Cluster analysis — Wikipedia, the free encyclopedia, 2018. URL https://en.wikipedia.org/w/index.php?title=Cluster_analysis&oldid=861063771.

[24] Davide Maltoni. Density-based approaches. URL http://bias.csr.unibo.it/maltoni/ml/DispensePDF/6_ML_Clustering.pdf. University of Bologna, Dipartimento di Informatica - Scienza e Ingegneria.

[25] H. Alpaydin. *Introduction to Machine Learning.* The MIT Press, Massachusetts, 1973. pp. 163-167, 176-179.

[26] Peter Flach. *Machine Learning.* Cambridge University Press, 2012.

[27] Michael R. Anderberg. *Cluster Analysis for Applications.* Academic Press, 1973.

[28] T. Caliński and J Harabasz. A dendrite method for cluster analysis. *Communications in Statistics*, 3(1):1–27, 1974. doi: 10.1080/03610927408827101.

[29] Silke Wagner and Dorothea Wagner. Comparing clusterings - an overview. Technical Report 4, Karlsruhe, 2007.

[30] Meenakshi Agarwal. Python tutorial – learn python programming step by step. URL http://www.techbeamers.com/python-tutorial-step-by-step/.

[31] Alessandro Micarelli. Machine learning: Classificazione e predizione, 2008. URL http://www.dia.uniroma3.it/~ia/docs/old/Classificazione%20e%20Predizione.pdf. Università Roma Tre.

[32] Unsupervised Feature Learning and Stanford Deep Learning. Data preprocessing. URL https://www.cs.utah.edu/~piyush/teaching/EM_algorithm.pdf.

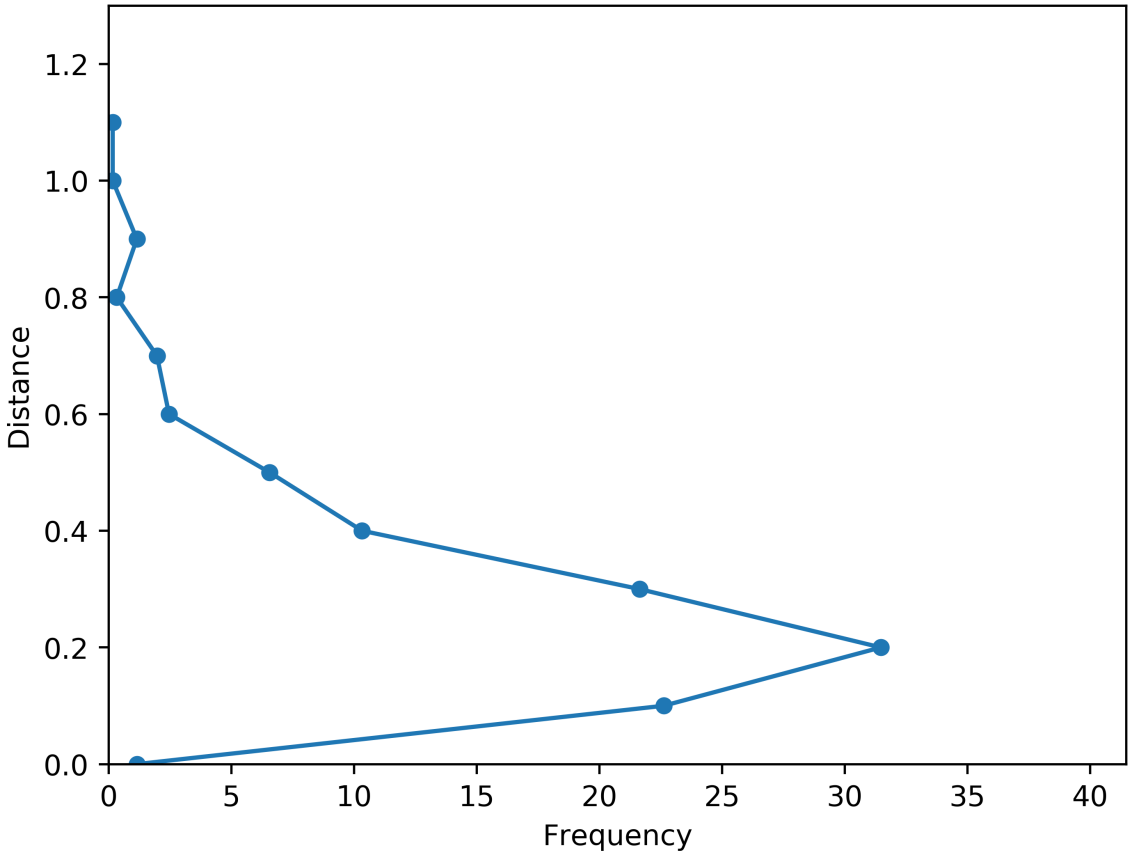
# Appendix A

# Automatic clustering results

Chosen hours: from 9 to 18
Chosen days: from 01/03/2018 to 30/04/2018
The coordinates chosen are: [[41.325, 2.125], [41.325, 2.175], [41.325, 2.225], [41.325, 2.275], [41.35, 2.1], [41.35, 2.15], [41.35, 2.2], [41.35, 2.25], [41.375, 2.125], [41.375, 2.175], [41.375, 2.225], [41.375, 2.275], [41.4, 2.1], [41.4, 2.15], [41.4, 2.2], [41.4, 2.25]]
Elapsed time to load 3660 files: 22.182 s
Dataset has 610 elements
Elapsed time to normalize data with time average speed was: 0.306 s
Elapsed time to normalize data with space average speed was: 0.280 s

## A.1 Automatic clustering infos

Completion of hierarchical clustering took 52.696 s
The calculation of the quality measures required 160.392 s
The best model was found in 156.867 s and is obtained from the measure GWB with gamma = 0.10
Assigning all the elements out of gamma took 0.280
The best model is composed of 17 clusters




The threshold chosen is: 0.60

## A.2 K-means infos

Execution time of k-means: 6.707 s
Number of iterations in k-means: 20
Elements that have changed cluster: 371
Number of outliers with distance greater than threshold: 17
Number of outliers with their own cluster: 0

## A.3 Clustering Results

Number of clusters: 17

Table A.1 Clusters information

| Cluster | N° elements | Relative freq | Min temp | Avg temp | Max temp | Min humidity | Avg humidity | Max humidity |
|---|---|---|---|---|---|---|---|---|
| 1 | 87 | 14.26 | 10.85 | 14.68 | 22.55 | 9.46 | 60.20 | 100.00 |
| 2 | 68 | 11.15 | 10.90 | 14.36 | 22.76 | 0.00 | 60.36 | 100.00 |
| 3 | 60 | 9.84 | 9.85 | 14.14 | 20.73 | 0.76 | 45.89 | 100.00 |
| 4 | 55 | 9.02 | 7.66 | 14.16 | 20.39 | 10.00 | 36.47 | 94.07 |
| 5 | 37 | 6.07 | 6.69 | 12.09 | 18.30 | 2.23 | 51.06 | 100.00 |
| 6 | 37 | 6.07 | 9.52 | 13.86 | 22.66 | 0.14 | 60.47 | 100.00 |
| 7 | 35 | 5.74 | 8.24 | 14.24 | 19.42 | 11.87 | 45.38 | 96.36 |
| 8 | 34 | 5.57 | 12.19 | 15.44 | 21.57 | 27.13 | 65.10 | 100.00 |
| 9 | 30 | 4.92 | 9.04 | 14.62 | 21.99 | 5.23 | 58.14 | 100.00 |
| 10 | 28 | 4.59 | 11.39 | 14.45 | 22.27 | 1.84 | 69.49 | 100.01 |
| 11 | 26 | 4.26 | 12.46 | 13.79 | 19.75 | 34.85 | 68.34 | 100.00 |
| 12 | 25 | 4.10 | 11.06 | 14.55 | 22.11 | 1.80 | 62.09 | 100.00 |
| 13 | 21 | 3.44 | 10.77 | 14.56 | 22.30 | 9.89 | 44.60 | 99.99 |
| 14 | 19 | 3.11 | 7.38 | 11.86 | 19.79 | 16.72 | 72.43 | 100.00 |
| 15 | 16 | 2.62 | 11.26 | 13.95 | 19.96 | 17.19 | 62.45 | 100.00 |
| 16 | 12 | 1.97 | 9.16 | 14.96 | 21.78 | 32.09 | 74.03 | 99.94 |
| 17 | 3 | 0.49 | 11.88 | 12.73 | 14.55 | 34.83 | 56.22 | 88.25 |
| ***18*** | 17 | 2.79 | 9.05 | 13.17 | 20.34 | 22.28 | 61.43 | 99.99 |

Table A.2 Clusters information

| Cluster | Min precipit | Avg precipit | Max Precipit | Min pressure | Avg pressure | Max pressure |
|---|---|---|---|---|---|---|
| 1 | -0.000 | 0.206 | 14.484 | 99632.469 | 101125.213 | 102277.070 |
| 2 | -0.000 | 1.144 | 17.891 | 99257.531 | 101100.968 | 102102.500 |
| 3 | -0.000 | 1.390 | 15.189 | 99154.547 | 100501.311 | 102020.406 |
| 4 | -0.000 | 0.823 | 18.596 | 99179.609 | 100482.714 | 102220.953 |
| 5 | -0.000 | 6.632 | 51.805 | 99198.844 | 100275.451 | 102095.203 |
| 6 | 0.000 | 1.816 | 19.094 | 99368.688 | 101201.785 | 102101.609 |
| 7 | -0.000 | 0.557 | 14.990 | 99143.766 | 100376.587 | 101732.688 |
| 8 | 0.000 | 0.023 | 1.095 | 100927.281 | 101804.244 | 102610.562 |
| 9 | -0.000 | 1.095 | 20.266 | 99274.312 | 101221.646 | 102625.109 |
| 10 | -0.000 | 1.688 | 19.828 | 99339.422 | 101161.523 | 102607.688 |
| 11 | 0.000 | 0.930 | 5.306 | 100335.125 | 101236.877 | 102582.031 |
| 12 | 0.000 | 1.015 | 19.398 | 99233.812 | 100983.363 | 102158.625 |
| 13 | -0.000 | 0.223 | 6.960 | 99262.938 | 100626.662 | 102206.109 |
| 14 | 0.000 | 3.881 | 54.170 | 99217.750 | 101028.177 | 102100.688 |
| 15 | -0.000 | 0.831 | 7.690 | 100006.344 | 101393.705 | 102571.203 |
| 16 | 0.000 | 0.035 | 1.325 | 101202.484 | 102061.446 | 102382.430 |
| 17 | 0.000 | 0.092 | 0.295 | 99506.375 | 100436.974 | 101011.531 |
| ***18*** | 0.000 | 0.909 | 11.327 | 100323.766 | 101491.046 | 101993.625 |

Table A.3 Clusters information

| Cluster | Min speed | Avg speed | Max speed | Min direction | Avg direction | Max direction |
|---|---|---|---|---|---|---|
| 1 | 0.331 | 5.209 | 13.965 | 0.223 | 31.515 | 359.858 |
| 2 | 0.490 | 4.577 | 13.846 | 0.225 | 17.803 | 359.845 |
| 3 | 0.950 | 6.444 | 14.102 | 6.859 | 49.433 | 340.346 |
| 4 | 0.548 | 5.581 | 13.692 | 0.140 | 97.443 | 354.869 |
| 5 | 0.410 | 6.131 | 15.556 | 56.981 | 122.596 | 224.414 |
| 6 | 0.351 | 3.720 | 8.548 | 0.049 | 348.575 | 359.956 |
| 7 | 0.433 | 5.989 | 13.032 | 7.676 | 68.011 | 351.089 |
| 8 | 0.139 | 2.723 | 6.704 | 158.937 | 251.012 | 348.742 |
| 9 | 0.818 | 3.137 | 7.650 | 5.959 | 277.988 | 328.510 |
| 10 | 0.209 | 2.906 | 7.783 | 6.310 | 307.702 | 357.459 |
| 11 | 1.936 | 7.190 | 12.748 | 213.308 | 236.735 | 278.402 |
| 12 | 0.461 | 2.791 | 6.165 | 0.867 | 327.017 | 359.691 |
| 13 | 0.353 | 5.538 | 12.237 | 1.521 | 50.022 | 356.641 |
| 14 | 0.345 | 4.982 | 17.646 | 47.606 | 204.986 | 296.768 |
| 15 | 0.348 | 4.949 | 12.987 | 72.577 | 253.093 | 338.555 |
| 16 | 0.283 | 1.347 | 3.093 | 184.303 | 275.532 | 355.433 |
| 17 | 0.145 | 1.221 | 2.509 | 1.855 | 45.860 | 357.504 |
| ***18*** | 0.166 | 2.458 | 15.499 | 3.520 | 258.872 | 359.755 |

Table A.4 Ranges of TEMPERATURE (in °C):

| Cluster | [6.687, 9.902) | [9.902, 13.117) | [13.117, 16.331) | [16.331, 19.546) | [19.546, 22.761] |
|---|---|---|---|---|---|
| 1 | 0.00% | 31.18% | 49.50% | 14.08% | 5.24% |
| 2 | 0.00% | 37.04% | 45.68% | 13.42% | 3.86% |
| 3 | 0.21% | 29.79% | 54.06% | 14.79% | 1.15% |
| 4 | 5.23% | 26.48% | 52.84% | 14.89% | 0.57% |
| 5 | 18.92% | 49.66% | 26.01% | 5.41% | 0.00% |
| 6 | 0.17% | 47.97% | 39.86% | 8.95% | 3.04% |
| 7 | 0.18% | 30.00% | 54.11% | 15.71% | 0.00% |
| 8 | 0.00% | 18.20% | 59.01% | 14.15% | 8.64% |
| 9 | 0.21% | 39.17% | 35.62% | 19.17% | 5.83% |
| 10 | 0.00% | 26.79% | 50.45% | 20.09% | 2.68% |
| 11 | 0.00% | 15.62% | 82.69% | 1.44% | 0.24% |
| 12 | 0.00% | 38.25% | 45.00% | 9.00% | 7.75% |
| 13 | 0.00% | 25.60% | 55.36% | 13.10% | 5.95% |
| 14 | 14.14% | 62.17% | 21.05% | 2.30% | 0.33% |
| 15 | 0.00% | 49.22% | 40.62% | 8.59% | 1.56% |
| 16 | 4.69% | 9.90% | 57.81% | 23.44% | 4.17% |
| 17 | 0.00% | 68.75% | 31.25% | 0.00% | 0.00% |
| ***18*** | 0.37% | 62.13% | 29.41% | 6.62% | 1.47% |

Table A.5 Ranges of HUMIDITY (in %):

| Cluster | [0.000, 20.001) | [20.001, 40.002) | [40.002, 60.004) | [60.004, 80.005) | [80.005, 100.006] |
|---|---|---|---|---|---|
| 1 | 5.17% | 19.61% | 26.22% | 22.77% | 26.22% |
| 2 | 7.63% | 21.69% | 22.70% | 14.06% | 33.92% |
| 3 | 23.85% | 24.90% | 20.83% | 13.33% | 17.08% |
| 4 | 24.09% | 43.41% | 13.64% | 14.43% | 4.43% |
| 5 | 19.59% | 18.24% | 17.06% | 24.32% | 20.78% |
| 6 | 8.11% | 20.95% | 17.40% | 14.19% | 39.36% |
| 7 | 18.75% | 29.82% | 21.07% | 23.75% | 6.61% |
| 8 | 0.00% | 29.41% | 11.95% | 24.08% | 34.56% |
| 9 | 16.67% | 22.50% | 11.25% | 18.75% | 30.83% |
| 10 | 3.57% | 5.36% | 22.54% | 29.46% | 39.06% |
| 11 | 0.00% | 8.41% | 29.09% | 27.40% | 35.10% |
| 12 | 4.00% | 14.50% | 32.00% | 18.25% | 31.25% |
| 13 | 22.02% | 28.57% | 14.88% | 17.26% | 17.26% |
| 14 | 1.97% | 9.87% | 5.59% | 41.45% | 41.12% |
| 15 | 3.12% | 27.73% | 14.06% | 21.48% | 33.59% |
| 16 | 0.00% | 2.60% | 8.85% | 61.98% | 26.56% |
| 17 | 0.00% | 33.33% | 33.33% | 10.42% | 22.92% |
| ***18*** | 0.00% | 25.37% | 14.34% | 35.66% | 24.63% |

Table A.6 Ranges of PRECIPITATION (in Kg/mˆ2):

| Cluster | [-0.000, 10.834) | [10.834, 21.668) | [21.668, 32.502) | [32.502, 43.336) | [43.336, 54.170] |
|---|---|---|---|---|---|
| 1 | 99.57% | 0.43% | 0.00% | 0.00% | 0.00% |
| 2 | 94.58% | 5.42% | 0.00% | 0.00% | 0.00% |
| 3 | 96.15% | 3.85% | 0.00% | 0.00% | 0.00% |
| 4 | 97.39% | 2.61% | 0.00% | 0.00% | 0.00% |
| 5 | 83.28% | 2.53% | 5.41% | 6.25% | 2.53% |
| 6 | 91.89% | 8.11% | 0.00% | 0.00% | 0.00% |
| 7 | 98.75% | 1.25% | 0.00% | 0.00% | 0.00% |
| 8 | 100.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 9 | 93.33% | 6.67% | 0.00% | 0.00% | 0.00% |
| 10 | 96.43% | 3.57% | 0.00% | 0.00% | 0.00% |
| 11 | 100.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 12 | 96.00% | 4.00% | 0.00% | 0.00% | 0.00% |
| 13 | 100.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 14 | 94.74% | 0.00% | 0.66% | 1.64% | 2.96% |
| 15 | 100.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 16 | 100.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 17 | 100.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| ***18*** | 99.63% | 0.37% | 0.00% | 0.00% | 0.00% |

Table A.7 Ranges of PRESSURE (in Pa):

| Cluster | [991k, 998k) | [998k, 1005k) | [1005k, 1012k) | [1012k, 1019k) | [1019k, 102625k] |
|---|---|---|---|---|---|
| 1 | 2.30% | 17.31% | 38.94% | 25.36% | 16.09% |
| 2 | 5.88% | 11.76% | 38.51% | 30.51% | 13.33% |
| 3 | 16.67% | 41.67% | 23.85% | 14.48% | 3.33% |
| 4 | 15.91% | 41.59% | 31.59% | 3.64% | 7.27% |
| 5 | 27.03% | 40.54% | 21.62% | 8.11% | 2.70% |
| 6 | 8.11% | 10.81% | 29.73% | 34.12% | 17.23% |
| 7 | 17.32% | 42.86% | 31.25% | 8.57% | 0.00% |
| 8 | 0.00% | 0.00% | 18.38% | 49.26% | 32.35% |
| 9 | 6.67% | 16.67% | 24.58% | 38.54% | 13.54% |
| 10 | 7.14% | 3.57% | 53.57% | 22.10% | 13.62% |
| 11 | 0.00% | 3.85% | 46.15% | 42.31% | 7.69% |
| 12 | 28.00% | 0.00% | 24.00% | 31.00% | 17.00% |
| 13 | 14.29% | 42.86% | 28.57% | 0.00% | 14.29% |
| 14 | 15.79% | 0.00% | 36.84% | 36.84% | 10.53% |
| 15 | 0.00% | 18.75% | 25.00% | 25.00% | 31.25% |
| 16 | 0.00% | 0.00% | 8.33% | 18.23% | 73.44% |
| 17 | 33.33% | 0.00% | 66.67% | 0.00% | 0.00% |
| ***18*** | 0.00% | 5.88% | 23.53% | 57.72% | 12.87% |

Table A.8 Wind speed ranges (in m/s)

| Cluster | [0, 2) | [2, 4) | [4, 6) | [6, 8) | [8, 10) | [10, 12) | [12, 14) | [14, 16) | [16, 17.65] |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 5.17% | 26.44% | 33.84% | 24.64% | 7.76% | 1.58% | 0.57% | 0.00% | 0.00% |
| 2 | 3.40% | 38.97% | 40.72% | 12.50% | 3.12% | 0.55% | 0.74% | 0.00% | 0.00% |
| 3 | 2.81% | 17.71% | 26.35% | 25.21% | 17.08% | 8.65% | 2.08% | 0.10% | 0.00% |
| 4 | 6.02% | 23.64% | 29.43% | 24.55% | 10.23% | 4.77% | 1.36% | 0.00% | 0.00% |
| 5 | 10.47% | 26.86% | 17.91% | 16.05% | 11.15% | 8.95% | 5.24% | 3.38% | 0.00% |
| 6 | 13.18% | 52.53% | 23.14% | 8.61% | 2.53% | 0.00% | 0.00% | 0.00% | 0.00% |
| 7 | 6.79% | 20.00% | 27.14% | 20.36% | 15.00% | 10.18% | 0.54% | 0.00% | 0.00% |
| 8 | 29.41% | 58.46% | 11.40% | 0.74% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 9 | 17.08% | 64.79% | 9.79% | 8.33% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 10 | 26.34% | 55.58% | 10.27% | 7.81% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 11 | 0.24% | 17.07% | 18.03% | 34.62% | 4.57% | 22.60% | 2.88% | 0.00% | 0.00% |
| 12 | 29.75% | 53.25% | 16.25% | 0.75% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 13 | 9.52% | 24.70% | 25.89% | 18.75% | 13.39% | 7.14% | 0.60% | 0.00% | 0.00% |
| 14 | 15.13% | 33.55% | 26.64% | 12.83% | 1.97% | 1.97% | 1.97% | 2.96% | 2.96% |
| 15 | 7.81% | 29.69% | 30.08% | 25.00% | 4.30% | 1.95% | 1.17% | 0.00% | 0.00% |
| 16 | 85.42% | 14.58% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 17 | 85.42% | 14.58% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| ***18*** | 59.56% | 25.74% | 6.62% | 2.57% | 1.47% | 2.57% | 1.10% | 0.37% | 0.00% |

Table A.9 Wind direction ranges

| Cluster | [, 45°) | [45°, 90°) | [90°, 135°) | [135°, 180°) | [180°, 225°) | [225°, 270°) | [270°, 315°) | [315°, 360°] |
|---|---|---|---|---|---|---|---|---|
| 1 | 94.83% | 4.02% | 0.22% | 0.00% | 0.00% | 0.00% | 0.00% | 0.93% |
| 2 | 92.83% | 0.74% | 0.09% | 0.00% | 0.00% | 0.00% | 0.09% | 6.25% |
| 3 | 33.02% | 66.56% | 0.21% | 0.00% | 0.00% | 0.00% | 0.00% | 0.21% |
| 4 | 1.36% | 22.73% | 75.34% | 0.45% | 0.00% | 0.00% | 0.00% | 0.11% |
| 5 | 0.00% | 1.18% | 75.17% | 23.48% | 0.17% | 0.00% | 0.00% | 0.00% |
| 6 | 40.20% | 0.17% | 0.00% | 0.00% | 0.00% | 0.00% | 0.84% | 58.78% |
| 7 | 1.96% | 84.29% | 13.57% | 0.00% | 0.00% | 0.00% | 0.00% | 0.18% |
| 8 | 0.00% | 0.00% | 0.00% | 0.18% | 10.11% | 71.32% | 16.73% | 1.65% |
| 9 | 0.21% | 0.21% | 0.00% | 0.00% | 0.00% | 25.00% | 72.71% | 1.88% |
| 10 | 0.22% | 0.22% | 0.00% | 0.00% | 0.00% | 0.89% | 69.20% | 29.46% |
| 11 | 0.00% | 0.00% | 0.00% | 0.00% | 6.73% | 93.03% | 0.24% | 0.00% |
| 12 | 3.25% | 0.25% | 0.25% | 0.00% | 0.00% | 0.50% | 12.25% | 83.50% |
| 13 | 50.30% | 37.50% | 11.90% | 0.00% | 0.00% | 0.00% | 0.00% | 0.30% |
| 14 | 0.00% | 0.33% | 1.97% | 27.96% | 66.12% | 2.96% | 0.66% | 0.00% |
| 15 | 0.00% | 0.78% | 0.00% | 3.52% | 3.12% | 85.16% | 6.64% | 0.78% |
| 16 | 0.00% | 0.00% | 0.00% | 0.00% | 9.90% | 36.98% | 45.83% | 7.29% |
| 17 | 39.58% | 20.83% | 6.25% | 6.25% | 0.00% | 2.08% | 0.00% | 25.00% |
| ***18*** | 18.75% | 11.76% | 10.66% | 15.07% | 10.66% | 5.88% | 15.44% | 11.76% |

Table A.10 Wind direction ranges

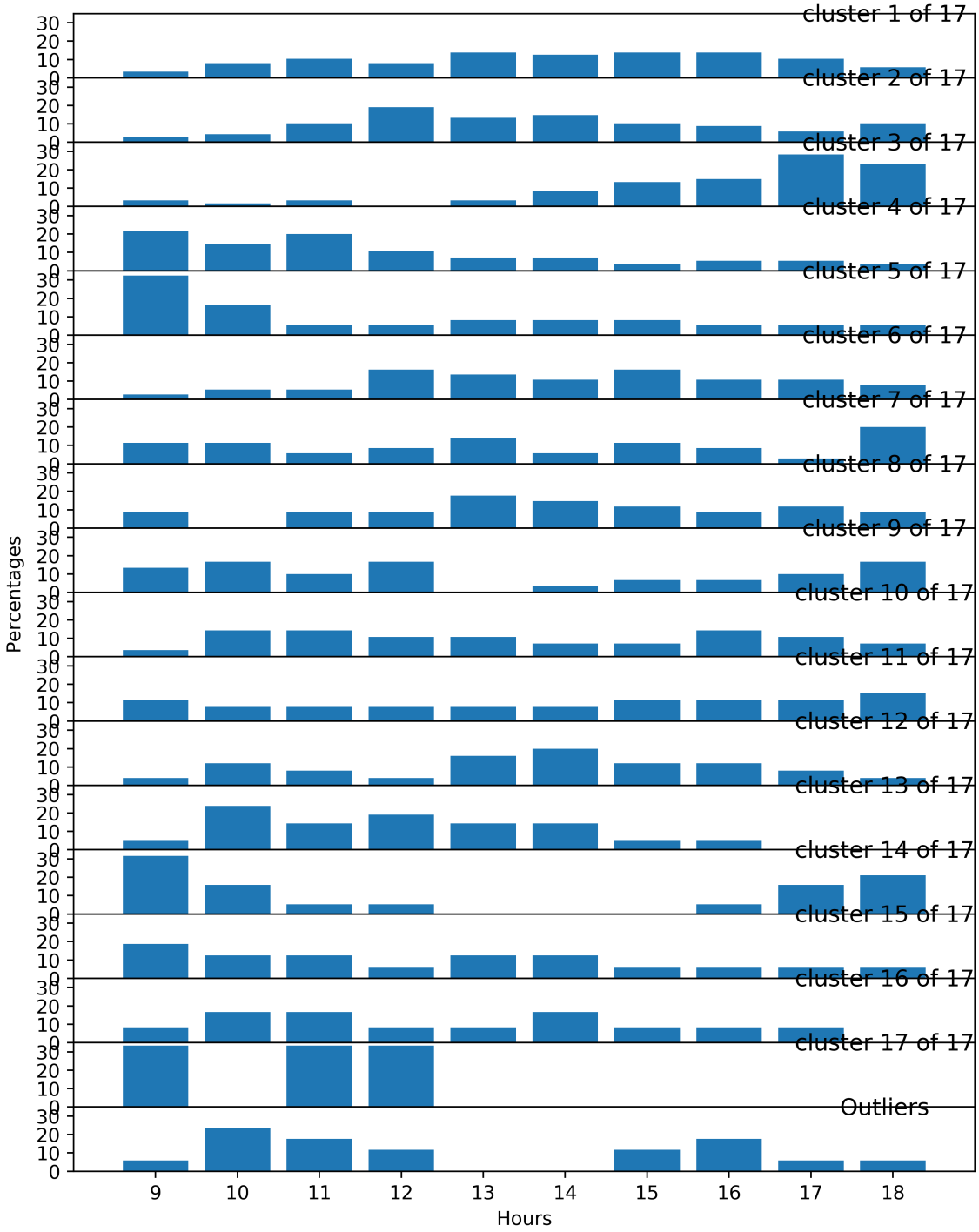
| Cluster | [0-22.5) | [22.5-45) | [45-67.5) | [67.5-90) | [90-112.5) | [112.5-135) | [135-157.5) | [157.5-180) | [180-202.5) | [202.5-225) | [225-247.5) | [247.5-270) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 21.12% | 73.71% | 3.52% | 0.50% | 0.22% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 2 | 73.62% | 19.21% | 0.55% | 0.18% | 0.09% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 3 | 1.35% | 31.67% | 65.83% | 0.73% | 0.21% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 4 | 0.34% | 1.02% | 2.84% | 19.89% | 66.59% | 8.75% | 0.34% | 0.11% | 0.00% | 0.00% | 0.00% | 0.00% |
| 5 | 0.00% | 0.00% | 0.17% | 1.01% | 16.39% | 58.78% | 19.09% | 4.39% | 0.00% | 0.17% | 0.00% | 0.00% |
| 6 | 37.84% | 2.36% | 0.17% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 7 | 0.18% | 1.79% | 48.04% | 36.25% | 13.39% | 0.18% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 8 | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.18% | 1.65% | 8.46% | 37.13% | 34.19% |
| 9 | 0.21% | 0.00% | 0.21% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 25.00% |
| 10 | 0.22% | 0.00% | 0.22% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.89% |
| 11 | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 6.73% | 84.62% | 8.41% |
| 12 | 2.75% | 0.50% | 0.00% | 0.25% | 0.25% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.50% |
| 13 | 5.06% | 45.24% | 24.11% | 13.39% | 10.71% | 1.19% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 14 | 0.00% | 0.00% | 0.33% | 0.00% | 0.33% | 1.64% | 8.88% | 19.08% | 39.80% | 26.32% | 2.30% | 0.66% |
| 15 | 0.00% | 0.00% | 0.00% | 0.78% | 0.00% | 0.00% | 2.73% | 0.78% | 0.00% | 3.12% | 21.48% | 63.67% |
| 16 | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 1.56% | 8.33% | 14.58% | 22.40% |
| 17 | 27.08% | 12.50% | 10.42% | 10.42% | 4.17% | 2.08% | 4.17% | 2.08% | 0.00% | 0.00% | 2.08% | 0.00% |
| ***18*** | 6.62% | 12.13% | 6.25% | 5.51% | 2.94% | 7.72% | 8.46% | 6.62% | 1.47% | 9.19% | 2.21% | 3.68% |

Table A.11 Wind direction wider ranges

| Cluster | [270-292.5) | [292.5-315) | [315-337.5) | [337.5-360] |
|---|---|---|---|---|
| 1 | 0.00% | 0.00% | 0.22% | 0.72% |
| 2 | 0.09% | 0.00% | 0.37% | 5.88% |
| 3 | 0.00% | 0.00% | 0.00% | 0.21% |
| 4 | 0.00% | 0.00% | 0.00% | 0.11% |
| 5 | 0.00% | 0.00% | 0.00% | 0.00% |
| 6 | 0.00% | 0.84% | 5.41% | 53.38% |
| 7 | 0.00% | 0.00% | 0.00% | 0.18% |
| 8 | 13.97% | 2.76% | 1.29% | 0.37% |
| 9 | 60.83% | 11.88% | 1.88% | 0.00% |
| 10 | 13.39% | 55.80% | 27.23% | 2.23% |
| 11 | 0.24% | 0.00% | 0.00% | 0.00% |
| 12 | 1.50% | 10.75% | 58.25% | 25.25% |
| 13 | 0.00% | 0.00% | 0.00% | 0.30% |
| 14 | 0.33% | 0.33% | 0.00% | 0.00% |
| 15 | 6.25% | 0.39% | 0.39% | 0.39% |
| 16 | 22.40% | 23.44% | 5.21% | 2.08% |
| 17 | 0.00% | 0.00% | 6.25% | 18.75% |
| **18*** | 7.35% | 8.09% | 6.99% | 4.78% |

Table A.12 Transition matrix

| Cluster | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 60.98 | 56.25 | 34.38 | 3.12 | 0.00 | 3.12 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 3.12 | 0.00 | 0.00 | 0.00 | 0.00 |
| 2 | 57.58 | 45.90 | 3.03 | 0.00 | 0.00 | 36.36 | 3.03 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 3 | 21.43 | 21.43 | 69.57 | 0.00 | 0.00 | 0.00 | 50.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 7.14 | 0.00 | 0.00 | 0.00 | 0.00 |
| 4 | 8.33 | 8.33 | 8.33 | 54.72 | 16.67 | 0.00 | 33.33 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 25.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 5 | 6.25 | 0.00 | 6.25 | 43.75 | 54.29 | 0.00 | 6.25 | 0.00 | 0.00 | 0.00 | 0.00 | 6.25 | 6.25 | 12.50 | 6.25 | 0.00 | 6.25 |
| 6 | 10.53 | 57.89 | 0.00 | 0.00 | 0.00 | 42.42 | 0.00 | 0.00 | 0.00 | 5.26 | 0.00 | 26.32 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 7 | 15.79 | 5.26 | 36.84 | 26.32 | 0.00 | 0.00 | 32.14 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 15.79 | 0.00 | 0.00 | 0.00 | 0.00 |
| 8 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 70.97 | 11.11 | 11.11 | 44.44 | 0.00 | 0.00 | 11.11 | 0.00 | 22.22 | 0.00 |
| 9 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 53.85 | 48.00 | 38.46 | 0.00 | 7.69 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 10 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 16.67 | 0.00 | 0.00 | 41.67 | 53.85 | 0.00 | 33.33 | 0.00 | 0.00 | 0.00 | 8.33 | 0.00 |
| 11 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 86.36 | 0.00 | 0.00 | 33.33 | 66.67 | 0.00 | 0.00 |
| 12 | 0.00 | 10.00 | 0.00 | 0.00 | 0.00 | 40.00 | 0.00 | 0.00 | 30.00 | 20.00 | 0.00 | 56.52 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 13 | 30.77 | 0.00 | 30.77 | 0.00 | 0.00 | 7.69 | 30.77 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 38.10 | 0.00 | 0.00 | 0.00 | 0.00 |
| 14 | 0.00 | 0.00 | 0.00 | 20.00 | 20.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 58.33 | 40.00 | 20.00 | 0.00 |
| 15 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 66.67 | 16.67 | 0.00 | 0.00 | 0.00 | 16.67 | 53.85 | 0.00 | 0.00 |
| 16 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 25.00 | 25.00 | 50.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 60.00 | 0.00 |
| 17 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 66.67 | 0.00 | 0.00 | 0.00 | 33.33 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |


cluster 1 of 17
cluster 2 of 17
cluster 3 of 17
cluster 4 of 17
cluster 5 of 17
cluster 6 of 17
cluster 7 of 17
cluster 8 of 17
cluster 9 of 17
cluster 10 of 17
cluster 11 of 17
cluster 12 of 17
cluster 13 of 17
cluster 14 of 17
cluster 15 of 17
cluster 16 of 17
cluster 17 of 17
Outliers
Percentages
0
10
20
30
9
10
11
12
13
14
15
16
17
18
Hours

# Appendix B

# Manual clustering results

Chosen hours: from 9 to 18
Chosen days: from 01/03/2018 to 30/04/2018
The coordinates chosen are: [[41.325, 2.125], [41.325, 2.175], [41.325, 2.225], [41.325, 2.275], [41.35, 2.1], [41.35, 2.15], [41.35, 2.2], [41.35, 2.25], [41.375, 2.125], [41.375, 2.175], [41.375, 2.225], [41.375, 2.275], [41.4, 2.1], [41.4, 2.15], [41.4, 2.2], [41.4, 2.25]]
Elapsed time to load 3660 files: 22.182 s
Dataset has 610 elements
Elapsed time to normalize data with time average speed was: 0.306 s
Elapsed time to normalize data with space average speed was: 0.280 s

## B.1 Manual clustering infos

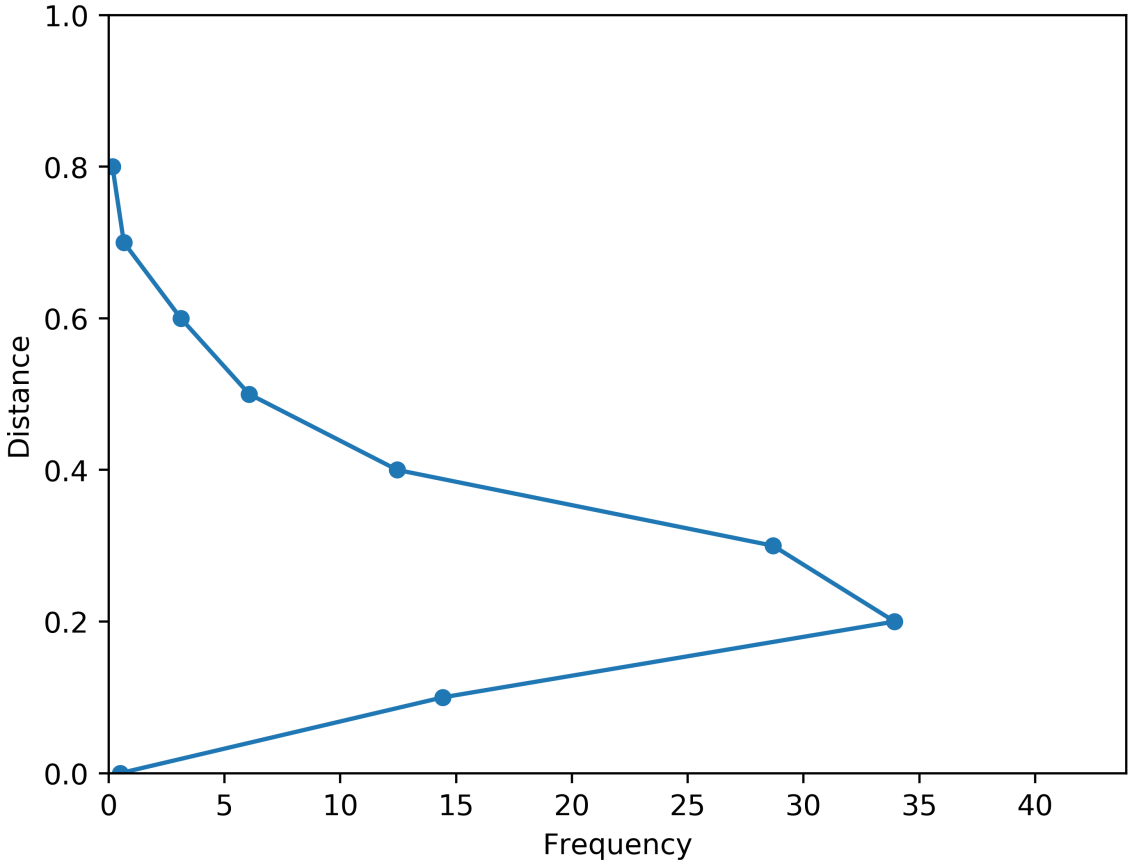
Generation of complete dendrogramThe hierarchical clustering algorithm execution time was 70.385Hierarchical clustering for k values chosenThe hierarchical clustering algorithm execution time was 54.963

## B.2 K-means infos

The chosen k are: **[17, 10]**

## B.3 Manual clustering results, k = 17

### B.3.1 Threshold




The threshold chosen is: 0.60

Execution time of k-means: 2.095 s
Number of iterations in k-means: 9
Elements that have changed cluster: 144
Number of outliers with distance greater than threshold: 10
Number of outliers with their own cluster: 3
Execution time of k-means: 3.219 s
Number of iterations in k-means: 14
Elements that have changed cluster: 250
Number of outliers with distance greater than threshold: 34
Number of outliers with their own cluster: 1

Number of clusters: 14

Table B.1 Clusters information

| Cluster | N° elements | Relative freq | Min temp | Avg temp | Max temp | Min humidity | Avg humidity | Max humidity |
|---|---|---|---|---|---|---|---|---|
| 1 | 135 | 22.13 | 10.85 | 14.58 | 22.76 | 6.63 | 60.90 | 100.00 |
| 2 | 99 | 16.23 | 9.85 | 14.30 | 22.30 | 0.76 | 45.30 | 100.00 |
| 3 | 67 | 10.98 | 11.26 | 14.49 | 21.57 | 17.19 | 64.61 | 100.00 |
| 4 | 62 | 10.16 | 9.52 | 13.92 | 22.66 | 0.00 | 59.97 | 100.00 |
| 5 | 62 | 10.16 | 7.66 | 14.11 | 20.39 | 10.00 | 40.86 | 94.07 |
| 6 | 42 | 6.89 | 11.39 | 14.41 | 22.11 | 1.80 | 66.31 | 100.01 |
| 7 | 38 | 6.23 | 6.72 | 12.93 | 19.16 | 2.23 | 39.20 | 98.55 |
| 8 | 29 | 4.75 | 9.04 | 14.44 | 21.99 | 5.23 | 57.55 | 100.00 |
| 9 | 23 | 3.77 | 9.16 | 15.46 | 22.27 | 32.09 | 71.87 | 100.00 |
| 10 | 16 | 2.62 | 6.69 | 11.22 | 15.48 | 22.28 | 70.40 | 100.00 |
| 11 | 15 | 2.46 | 7.38 | 12.40 | 19.79 | 16.72 | 74.18 | 100.00 |
| 12 | 3 | 0.49 | 12.71 | 14.24 | 18.73 | 45.72 | 61.53 | 72.85 |
| 13 | 3 | 0.49 | 11.38 | 13.83 | 19.04 | 53.67 | 71.40 | 99.99 |
| 14 | 3 | 0.49 | 9.05 | 12.33 | 13.56 | 29.87 | 52.26 | 88.16 |
| ***15*** | 13 | 2.13 | 8.41 | 13.46 | 20.87 | 24.29 | 60.33 | 96.98 |

Table B.2 Clusters information

| Cluster | Min precipit | Avg precipit | Max Precipit | Min pressure | Avg pressure | Max pressure |
|---|---|---|---|---|---|---|
| 1 | -0.000 | 0.388 | 14.753 | 99632.469 | 101132.860 | 102256.227 |
| 2 | -0.000 | 0.895 | 15.189 | 99154.547 | 100536.878 | 102277.070 |
| 3 | 0.000 | 0.492 | 5.498 | 100313.234 | 101551.284 | 102610.562 |
| 4 | 0.000 | 1.786 | 19.094 | 99257.531 | 101112.127 | 102102.500 |
| 5 | -0.000 | 1.034 | 18.596 | 99143.766 | 100438.096 | 102220.953 |
| 6 | -0.000 | 1.725 | 19.828 | 99233.812 | 100943.501 | 102158.625 |
| 7 | -0.000 | 3.974 | 44.692 | 99198.844 | 100339.052 | 102095.203 |
| 8 | -0.000 | 1.313 | 20.266 | 99274.312 | 101131.641 | 102625.109 |
| 9 | 0.000 | 0.020 | 1.325 | 101202.484 | 102063.865 | 102607.688 |
| 10 | 0.000 | 5.920 | 51.805 | 99248.375 | 100600.985 | 101956.250 |
| 11 | -0.000 | 4.304 | 54.170 | 99285.156 | 101040.120 | 102100.688 |
| 12 | 0.000 | 0.000 | 0.000 | 101494.422 | 101578.993 | 101628.953 |
| 13 | -0.000 | 2.154 | 11.327 | 100006.344 | 100575.825 | 101369.391 |
| 14 | 0.000 | 2.558 | 10.426 | 100955.188 | 101275.212 | 101877.750 |
| ***15*** | 0.000 | 0.802 | 10.363 | 99217.750 | 101264.104 | 101993.625 |

Table B.3 Clusters information

| Cluster | Min speed | Avg speed | Max speed | Min direction | Avg direction | Max direction |
|---|---|---|---|---|---|---|
| 1 | 0.331 | 5.008 | 13.965 | 0.223 | 27.910 | 359.858 |
| 2 | 0.353 | 6.243 | 14.102 | 1.521 | 51.932 | 356.246 |
| 3 | 0.458 | 5.046 | 12.987 | 148.918 | 243.827 | 341.776 |
| 4 | 0.351 | 3.793 | 10.160 | 0.049 | 13.104 | 359.956 |
| 5 | 0.540 | 5.295 | 12.574 | 0.140 | 88.505 | 354.869 |
| 6 | 0.350 | 3.067 | 7.783 | 0.867 | 314.927 | 359.691 |
| 7 | 0.557 | 6.868 | 15.556 | 56.981 | 114.283 | 173.865 |
| 8 | 0.546 | 3.280 | 7.650 | 5.959 | 277.621 | 338.555 |
| 9 | 0.139 | 1.380 | 3.093 | 6.310 | 276.873 | 355.433 |
| 10 | 0.318 | 4.017 | 15.164 | 19.844 | 145.530 | 224.414 |
| 11 | 0.345 | 5.384 | 17.646 | 47.606 | 206.611 | 296.768 |
| 12 | 0.432 | 1.937 | 4.399 | 1.205 | 39.812 | 358.913 |
| 13 | 0.259 | 2.820 | 5.459 | 75.180 | 252.410 | 283.351 |
| 14 | 0.493 | 3.035 | 15.499 | 18.184 | 114.500 | 216.184 |
| ***15*** | 0.145 | 2.151 | 13.456 | 2.741 | 262.230 | 359.755 |

Table B.4 Ranges of TEMPERATURE (in °C):

| Cluster | [6.687, 9.902) | [9.902, 13.117) | [13.117, 16.331) | [16.331, 19.546) | [19.546, 22.761] |
|---|---|---|---|---|---|
| 1 | 0.00% | 33.94% | 47.27% | 13.89% | 4.91% |
| 2 | 0.13% | 27.08% | 56.00% | 14.90% | 1.89% |
| 3 | 0.00% | 24.07% | 64.46% | 7.46% | 4.01% |
| 4 | 0.10% | 45.67% | 41.63% | 8.97% | 3.63% |
| 5 | 4.64% | 28.43% | 51.41% | 15.02% | 0.50% |
| 6 | 0.00% | 31.25% | 48.66% | 16.67% | 3.42% |
| 7 | 15.46% | 36.02% | 38.98% | 9.54% | 0.00% |
| 8 | 0.22% | 42.67% | 33.41% | 18.53% | 5.17% |
| 9 | 2.45% | 8.15% | 60.05% | 21.47% | 7.88% |
| 10 | 15.23% | 79.30% | 5.47% | 0.00% | 0.00% |
| 11 | 7.92% | 60.42% | 28.33% | 2.92% | 0.42% |
| 12 | 0.00% | 45.83% | 33.33% | 20.83% | 0.00% |
| 13 | 0.00% | 52.08% | 37.50% | 10.42% | 0.00% |
| 14 | 2.08% | 89.58% | 8.33% | 0.00% | 0.00% |
| ***15*** | 1.92% | 47.60% | 39.90% | 6.73% | 3.85% |

Table B.5 Ranges of HUMIDITY (in %):

| Cluster | [0.000, 20.001) | [20.001, 40.002) | [40.002, 60.004) | [60.004, 80.005) | [80.005, 100.006] |
|---|---|---|---|---|---|
| 1 | 5.69% | 20.46% | 23.29% | 21.62% | 28.94% |
| 2 | 20.71% | 29.99% | 19.89% | 14.02% | 15.40% |
| 3 | 0.75% | 24.81% | 17.35% | 23.13% | 33.96% |
| 4 | 9.68% | 19.46% | 19.46% | 11.79% | 39.62% |
| 5 | 22.88% | 32.96% | 19.35% | 19.35% | 5.44% |
| 6 | 4.76% | 9.82% | 23.21% | 25.89% | 36.31% |
| 7 | 27.14% | 33.55% | 13.32% | 15.62% | 10.36% |
| 8 | 17.24% | 23.28% | 11.64% | 15.09% | 32.76% |
| 9 | 0.00% | 1.36% | 20.65% | 54.35% | 23.64% |
| 10 | 0.00% | 7.81% | 18.36% | 37.11% | 36.72% |
| 11 | 2.50% | 10.83% | 2.08% | 33.33% | 51.25% |
| 12 | 0.00% | 0.00% | 33.33% | 66.67% | 0.00% |
| 13 | 0.00% | 0.00% | 39.58% | 27.08% | 33.33% |
| 14 | 0.00% | 56.25% | 10.42% | 0.00% | 33.33% |
| ***15*** | 0.00% | 20.19% | 24.04% | 38.94% | 16.83% |

Table B.6 Ranges of PRECIPITATION (in Kg/m^2):

| Cluster | [-0.000, 10.834) | [10.834, 21.668) | [21.668, 32.502) | [32.502, 43.336) | [43.336, 54.170] |
|---|---|---|---|---|---|
| 1 | 99.21% | 0.79% | 0.00% | 0.00% | 0.00% |
| 2 | 97.66% | 2.34% | 0.00% | 0.00% | 0.00% |
| 3 | 100.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 4 | 90.32% | 9.68% | 0.00% | 0.00% | 0.00% |
| 5 | 96.98% | 3.02% | 0.00% | 0.00% | 0.00% |
| 6 | 95.24% | 4.76% | 0.00% | 0.00% | 0.00% |
| 7 | 89.14% | 2.30% | 4.61% | 3.78% | 0.16% |
| 8 | 93.10% | 6.90% | 0.00% | 0.00% | 0.00% |
| 9 | 100.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 10 | 87.11% | 0.39% | 1.56% | 5.47% | 5.47% |
| 11 | 93.33% | 0.00% | 0.83% | 2.08% | 3.75% |
| 12 | 100.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 13 | 97.92% | 2.08% | 0.00% | 0.00% | 0.00% |
| 14 | 100.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| ***15*** | 100.00% | 0.00% | 0.00% | 0.00% | 0.00% |

Table B.7 Ranges of PRESSURE (in Pa):

| Cluster | [991k, 998k) | [998k, 1005k) | [1005k, 1012k) | [1012k, 1019k) | [1019k, 102625k] |
|---|---|---|---|---|---|
| 1 | 2.22% | 14.86% | 41.53% | 27.64% | 13.75% |
| 2 | 14.20% | 43.43% | 25.51% | 10.80% | 6.06% |
| 3 | 0.00% | 2.99% | 31.72% | 41.42% | 23.88% |
| 4 | 11.29% | 9.68% | 32.26% | 29.23% | 17.54% |
| 5 | 18.55% | 42.14% | 28.02% | 4.84% | 6.45% |
| 6 | 19.05% | 2.38% | 42.86% | 26.04% | 9.67% |
| 7 | 19.08% | 44.08% | 28.95% | 5.26% | 2.63% |
| 8 | 6.90% | 20.69% | 25.43% | 36.42% | 10.56% |
| 9 | 0.00% | 0.00% | 4.35% | 31.25% | 64.40% |
| 10 | 31.25% | 12.50% | 31.25% | 18.75% | 6.25% |
| 11 | 13.33% | 0.00% | 40.00% | 33.33% | 13.33% |
| 12 | 0.00% | 0.00% | 0.00% | 100.00% | 0.00% |
| 13 | 0.00% | 66.67% | 0.00% | 33.33% | 0.00% |
| 14 | 0.00% | 0.00% | 66.67% | 33.33% | 0.00% |
| ***15*** | 15.38% | 0.00% | 15.38% | 60.10% | 9.13% |

Table B.8 Wind speed ranges (in m/s)

| Cluster | [0, 2) | [2, 4) | [4, 6) | [6, 8) | [8, 10) | [10, 12) | [12, 14) | [14, 16) | [16, 17.65] |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 4.26% | 29.86% | 37.59% | 21.39% | 4.95% | 1.20% | 0.74% | 0.00% | 0.00% |
| 2 | 4.29% | 19.51% | 26.58% | 22.29% | 16.79% | 8.90% | 1.58% | 0.06% | 0.00% |
| 3 | 8.49% | 40.95% | 17.82% | 19.31% | 2.80% | 9.24% | 1.40% | 0.00% | 0.00% |
| 4 | 13.10% | 51.11% | 23.79% | 7.66% | 4.13% | 0.20% | 0.00% | 0.00% | 0.00% |
| 5 | 8.17% | 25.10% | 29.33% | 22.88% | 9.98% | 4.13% | 0.40% | 0.00% | 0.00% |
| 6 | 20.39% | 58.48% | 15.48% | 5.65% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 7 | 4.77% | 17.60% | 21.71% | 22.37% | 14.64% | 11.51% | 5.26% | 2.14% | 0.00% |
| 8 | 12.72% | 66.38% | 11.42% | 9.48% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 9 | 86.41% | 13.59% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 10 | 24.22% | 43.75% | 15.62% | 6.64% | 1.56% | 2.73% | 2.73% | 2.73% | 0.00% |
| 11 | 11.67% | 30.00% | 31.67% | 12.50% | 1.67% | 2.50% | 2.50% | 3.75% | 3.75% |
| 12 | 56.25% | 39.58% | 4.17% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 13 | 33.33% | 39.58% | 27.08% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 14 | 52.08% | 22.92% | 10.42% | 8.33% | 2.08% | 2.08% | 0.00% | 2.08% | 0.00% |
| ***15*** | 71.63% | 19.23% | 1.44% | 0.96% | 2.40% | 2.88% | 1.44% | 0.00% | 0.00% |

Table B.9 Wind direction ranges

| Cluster | [, 45°) | [45°, 90°) | [90°, 135°) | [135°, 180°) | [180°, 225°) | [225°, 270°) | [270°, 315°) | [315°, 360°] |
|---|---|---|---|---|---|---|---|---|
| 1 | 95.32% | 2.87% | 0.42% | 0.00% | 0.00% | 0.00% | 0.00% | 1.39% |
| 2 | 32.39% | 64.02% | 3.41% | 0.00% | 0.00% | 0.00% | 0.00% | 0.19% |
| 3 | 0.00% | 0.00% | 0.00% | 0.28% | 5.50% | 86.38% | 7.37% | 0.47% |
| 4 | 50.20% | 0.60% | 0.10% | 0.00% | 0.00% | 0.00% | 0.60% | 48.49% |
| 5 | 1.71% | 41.94% | 55.75% | 0.40% | 0.00% | 0.00% | 0.00% | 0.20% |
| 6 | 0.45% | 0.15% | 0.00% | 0.00% | 0.00% | 0.30% | 45.68% | 53.42% |
| 7 | 0.00% | 2.30% | 95.07% | 2.63% | 0.00% | 0.00% | 0.00% | 0.00% |
| 8 | 0.22% | 0.43% | 0.00% | 0.22% | 0.00% | 26.08% | 70.91% | 2.16% |
| 9 | 0.27% | 0.00% | 0.00% | 0.00% | 8.15% | 32.61% | 51.36% | 7.61% |
| 10 | 1.17% | 0.78% | 16.02% | 78.52% | 3.52% | 0.00% | 0.00% | 0.00% |
| 11 | 0.00% | 0.42% | 0.42% | 10.00% | 83.75% | 4.58% | 0.83% | 0.00% |
| 12 | 31.25% | 10.42% | 2.08% | 0.00% | 4.17% | 6.25% | 14.58% | 31.25% |
| 13 | 0.00% | 4.17% | 8.33% | 16.67% | 16.67% | 37.50% | 16.67% | 0.00% |
| 14 | 20.83% | 20.83% | 29.17% | 20.83% | 8.33% | 0.00% | 0.00% | 0.00% |
| ***15*** | 20.19% | 10.58% | 6.25% | 7.21% | 13.46% | 7.21% | 13.46% | 21.63% |

Table B.10 Wind direction wider ranges

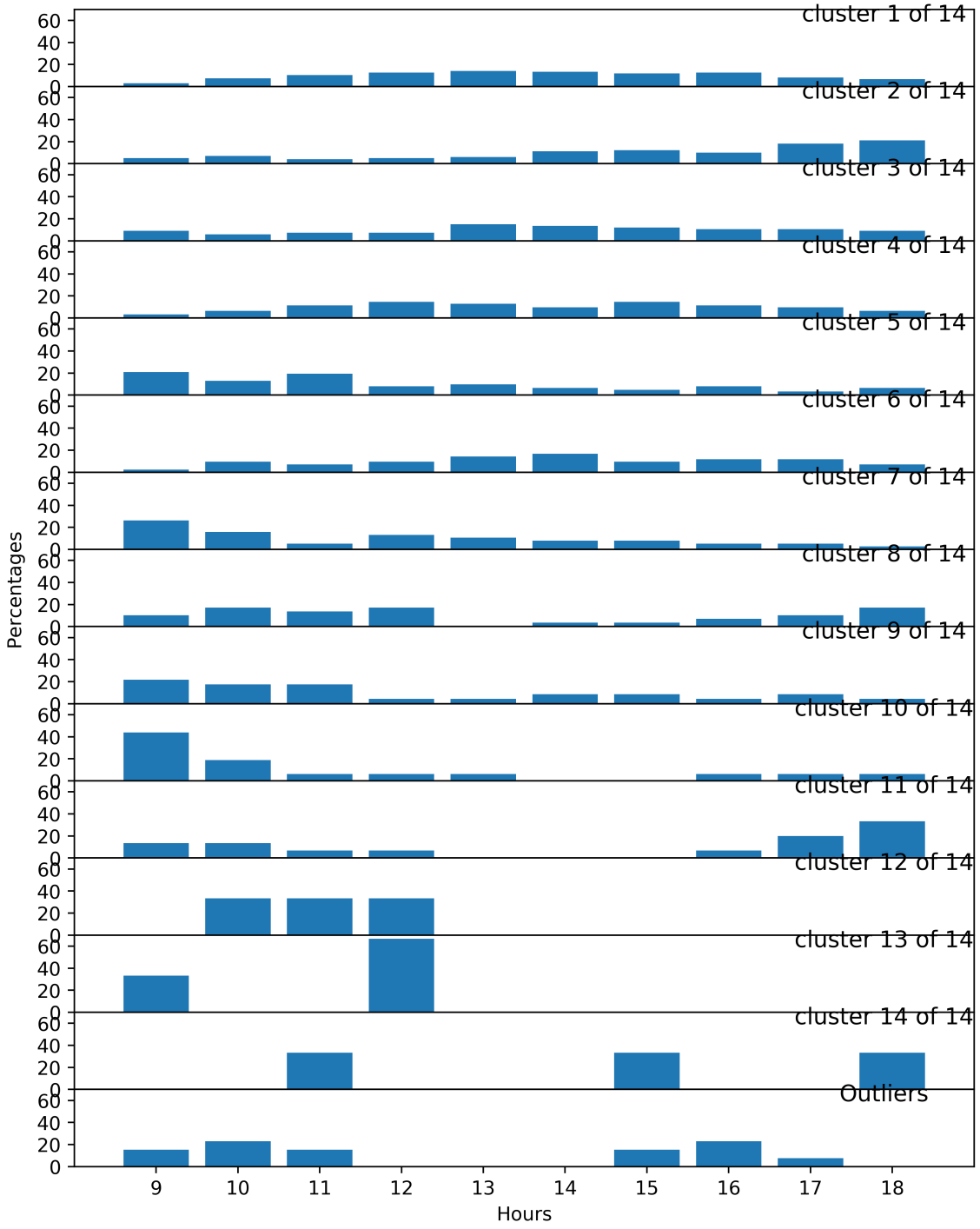
| Cluster | [0-22.5) | [22.5-45) | [45-67.5) | [67.5-90) | [90-112.5) | [112.5-135) | [135-157.5) | [157.5-180) | [180-202.5) | [202.5-225) | [225-247.5) | [247.5-270) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 40.51% | 54.81% | 2.36% | 0.51% | 0.37% | 0.05% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 2 | 1.83% | 30.56% | 56.88% | 7.13% | 3.22% | 0.19% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 3 | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.09% | 0.19% | 0.65% | 4.85% | 54.57% | 31.81% |
| 4 | 45.97% | 4.23% | 0.40% | 0.20% | 0.10% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 5 | 0.40% | 1.31% | 10.79% | 31.15% | 51.41% | 4.33% | 0.30% | 0.10% | 0.00% | 0.00% | 0.00% | 0.00% |
| 6 | 0.30% | 0.15% | 0.15% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.30% |
| 7 | 0.00% | 0.00% | 0.33% | 1.97% | 37.17% | 57.89% | 2.30% | 0.33% | 0.00% | 0.00% | 0.00% | 0.00% |
| 8 | 0.22% | 0.00% | 0.22% | 0.22% | 0.00% | 0.00% | 0.22% | 0.00% | 0.00% | 0.00% | 0.00% | 26.08% |
| 9 | 0.27% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 1.36% | 6.79% | 11.68% | 20.92% |
| 10 | 0.39% | 0.78% | 0.00% | 0.78% | 1.95% | 14.06% | 48.44% | 30.08% | 2.34% | 1.17% | 0.00% | 0.00% |
| 11 | 0.00% | 0.00% | 0.42% | 0.00% | 0.00% | 0.42% | 3.33% | 6.67% | 46.25% | 37.50% | 3.75% | 0.83% |
| 12 | 18.75% | 12.50% | 10.42% | 0.00% | 0.00% | 2.08% | 0.00% | 0.00% | 0.00% | 4.17% | 0.00% | 6.25% |
| 13 | 0.00% | 0.00% | 0.00% | 4.17% | 2.08% | 6.25% | 14.58% | 2.08% | 0.00% | 16.67% | 14.58% | 22.92% |
| 14 | 2.08% | 18.75% | 8.33% | 12.50% | 10.42% | 18.75% | 16.67% | 4.17% | 6.25% | 2.08% | 0.00% | 0.00% |
| ***15*** | 8.65% | 11.54% | 5.77% | 4.81% | 2.40% | 3.85% | 4.33% | 2.88% | 2.40% | 11.06% | 3.37% | 3.85% |

Table B.11 Wind direction wider ranges

| Cluster | [270-292.5) | [292.5-315) | [315-337.5) | [337.5-360] |
|---|---|---|---|---|
| 1 | 0.00% | 0.00% | 0.14% | 1.25% |
| 2 | 0.00% | 0.00% | 0.00% | 0.19% |
| 3 | 6.44% | 0.93% | 0.37% | 0.09% |
| 4 | 0.00% | 0.60% | 7.66% | 40.83% |
| 5 | 0.00% | 0.00% | 0.00% | 0.20% |
| 6 | 6.10% | 39.58% | 44.20% | 9.23% |
| 7 | 0.00% | 0.00% | 0.00% | 0.00% |
| 8 | 59.91% | 10.99% | 1.94% | 0.22% |
| 9 | 28.80% | 22.55% | 5.98% | 1.63% |
| 10 | 0.00% | 0.00% | 0.00% | 0.00% |
| 11 | 0.42% | 0.42% | 0.00% | 0.00% |
| 12 | 4.17% | 10.42% | 18.75% | 12.50% |
| 13 | 16.67% | 0.00% | 0.00% | 0.00% |
| 14 | 0.00% | 0.00% | 0.00% | 0.00% |
| ***15*** | 5.29% | 8.17% | 11.06% | 10.58 |

Table B.12 Transition matrix

| Cluster | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 80.16 | 52.00 | 0.00 | 40.00 | 8.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 2 | 55.56 | 76.92 | 0.00 | 5.56 | 33.33 | 0.00 | 5.56 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 3 | 0.00 | 0.00 | 81.97 | 0.00 | 0.00 | 0.00 | 0.00 | 45.45 | 9.09 | 0.00 | 36.36 | 0.00 | 0.00 | 9.09 |
| 4 | 57.14 | 4.76 | 0.00 | 63.79 | 0.00 | 33.33 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 4.76 |
| 5 | 25.00 | 57.14 | 0.00 | 0.00 | 51.72 | 0.00 | 17.86 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 6 | 0.00 | 0.00 | 0.00 | 50.00 | 0.00 | 63.16 | 0.00 | 42.86 | 7.14 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 7 | 6.25 | 12.50 | 0.00 | 0.00 | 62.50 | 0.00 | 56.76 | 0.00 | 0.00 | 18.75 | 0.00 | 0.00 | 0.00 | 0.00 |
| 8 | 0.00 | 0.00 | 38.46 | 0.00 | 0.00 | 38.46 | 0.00 | 45.83 | 15.38 | 0.00 | 0.00 | 0.00 | 7.69 | 0.00 |
| 9 | 0.00 | 0.00 | 25.00 | 0.00 | 0.00 | 37.50 | 0.00 | 37.50 | 60.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 10 | 0.00 | 10.00 | 10.00 | 10.00 | 0.00 | 0.00 | 10.00 | 0.00 | 10.00 | 23.08 | 30.00 | 0.00 | 10.00 | 10.00 |
| 11 | 0.00 | 0.00 | 66.67 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 33.33 | 62.50 | 0.00 | 0.00 | 0.00 |
| 12 | 0.00 | 0.00 | 0.00 | 100.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 66.67 | 0.00 | 0.00 |
| 13 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 33.33 | 0.00 | 33.33 | 0.00 | 33.33 | 0.00 | 0.00 | 0.00 | 0.00 |
| 14 | 0.00 | 0.00 | 0.00 | 100.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |


cluster 1 of 14
cluster 2 of 14
cluster 3 of 14
cluster 4 of 14
cluster 5 of 14
cluster 6 of 14
cluster 7 of 14
cluster 8 of 14
cluster 9 of 14
cluster 10 of 14
cluster 11 of 14
cluster 12 of 14
cluster 13 of 14
cluster 14 of 14
Outliers
Percentages
Hours
0
20
40
60
9
10
11
12
13
14
15
16
17
18

### B.3.2 Comparison with automatic clustering

Table B.13 Clusters matching

| Cluster automatic | Cluster manual k = 17 | Common elements |
|---|---|---|
| 1 (87) | 1 (135) | 84 |
| 3 (60) | 2 (99) | 60 |
| 4 (55) | 5 (62) | 44 |
| 6 (37) | 4 (62) | 37 |
| 8 (34) | 3 (67) | 29 |
| 9 (30) | 8 (29) | 28 |
| 5 (37) | 7 (38) | 27 |
| 10 (28) | 6 (42) | 24 |
| 14 (19) | 11 (15) | 14 |
| 16 (12) | 9 (23) | 12 |
| 15 (16) | 13 (3) | 2 |
| 2 (68) | 12 (3) | 1 |
| 7 (35) | 10 (16) | 0 |
| 11 (26) | 14 (3) | 0 |

Maximum Matching Measure = 0.593

### B.3.3 Comparison with manual clustering k = 10

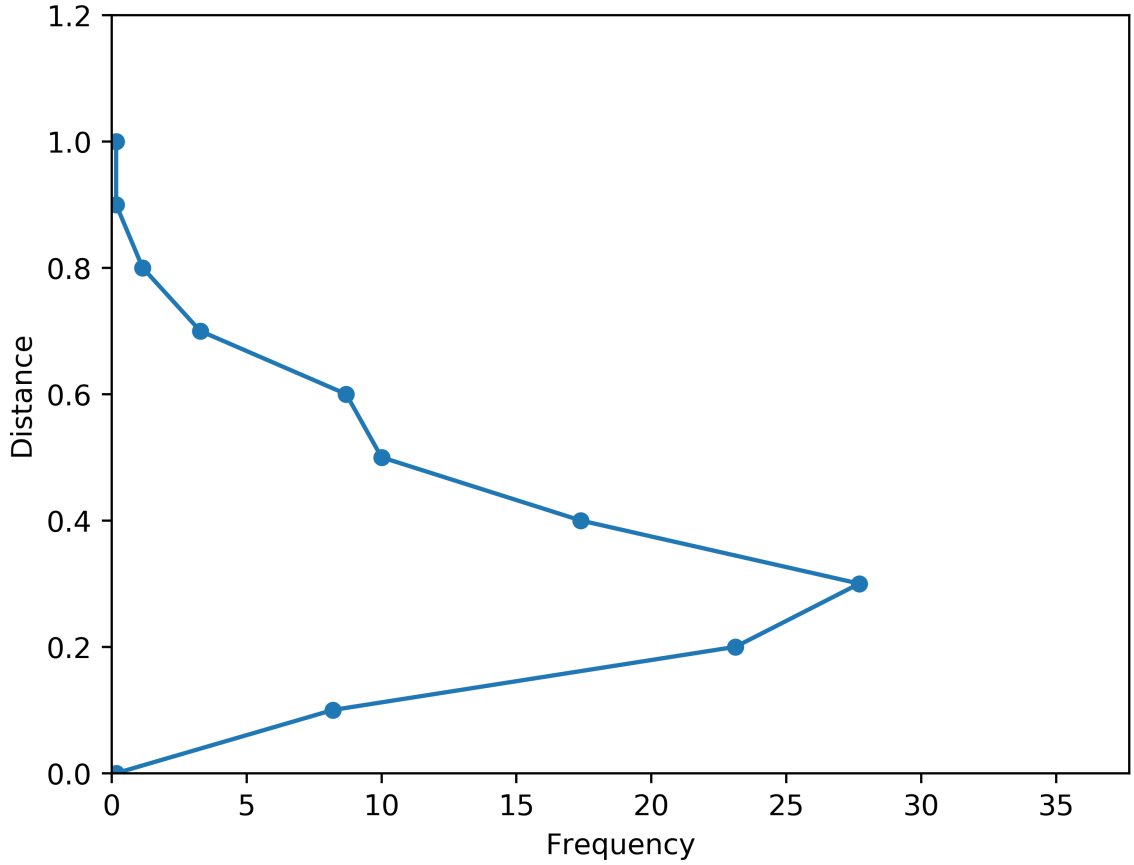
Table B.14 Clusters matching

| Cluster manual k = 17 | Cluster manual k = 10 | Common elements |
|---|---|---|
| 1 (135) | 9 (151) | 115 |
| 3 (67) | 8 (88) | 67 |
| 2 (99) | 1 (80) | 62 |
| 4 (62) | 7 (85) | 62 |
| 5 (62) | 6 (84) | 43 |
| 6 (42) | 2 (63) | 39 |
| 11 (15) | 4 (18) | 15 |
| 13 (3) | 3 (3) | 3 |
| 14 (3) | 5 (3) | 3 |

Maximum Matching Measure = 0.670

# B.4 Manual clustering results, k = 10

## B.4.1 Threshold




The threshold chosen is: 0.60

Execution time of k-means: 2.095 s
Number of iterations in k-means: 9
Elements that have changed cluster: 144
Number of outliers with distance greater than threshold: 10
Number of outliers with their own cluster: 3
Execution time of k-means: 3.219 s
Number of iterations in k-means: 14
Elements that have changed cluster: 250
Number of outliers with distance greater than threshold: 34
Number of outliers with their own cluster: 1

Number of clusters: 9

Table B.15 Clusters information

| Cluster | N° elements | Relative freq | Min temp | Avg temp | Max temp | Min humidity | Avg humidity | Max humidity |
|---|---|---|---|---|---|---|---|---|
| 1 | 151 | 24.75 | 10.65 | 14.63 | 22.76 | 9.46 | 58.72 | 100.00 |
| 2 | 88 | 14.43 | 9.04 | 14.51 | 21.92 | 9.18 | 63.95 | 100.00 |
| 3 | 85 | 13.93 | 9.52 | 14.04 | 22.66 | 0.00 | 60.52 | 100.00 |
| 4 | 84 | 13.77 | 6.69 | 13.49 | 20.39 | 2.23 | 39.57 | 100.00 |
| 5 | 80 | 13.11 | 8.24 | 14.12 | 22.30 | 0.76 | 42.27 | 100.00 |
| 6 | 63 | 10.33 | 11.17 | 14.69 | 22.27 | 1.80 | 64.35 | 100.01 |
| 7 | 18 | 2.95 | 7.38 | 12.10 | 19.79 | 16.72 | 72.22 | 100.00 |
| 8 | 3 | 0.49 | 11.38 | 13.83 | 19.04 | 53.67 | 71.40 | 99.99 |
| 9 | 3 | 0.49 | 9.05 | 12.33 | 13.56 | 29.87 | 52.26 | 88.16 |
| ***10*** | 35 | 5.74 | 7.21 | 13.23 | 20.87 | 22.28 | 68.34 | 100.00 |

Table B.16 Clusters information

| Cluster | Min precipit | Avg precipit | Max Precipit | Min pressure | Avg pressure | Max pressure |
|---|---|---|---|---|---|---|
| 1 | -0.000 | 0.542 | 15.189 | 99262.938 | 101016.159 | 102277.070 |
| 2 | -0.000 | 0.572 | 20.266 | 99274.312 | 101546.365 | 102625.109 |
| 3 | 0.000 | 1.501 | 19.094 | 99257.531 | 101126.782 | 102102.500 |
| 4 | -0.000 | 2.770 | 47.966 | 99179.609 | 100396.522 | 102220.953 |
| 5 | -0.000 | 0.767 | 15.189 | 99143.766 | 100387.576 | 102110.594 |
| 6 | -0.000 | 1.393 | 19.898 | 99267.688 | 101171.538 | 102607.688 |
| 7 | 0.000 | 3.605 | 54.170 | 99285.156 | 101112.146 | 102100.688 |
| 8 | -0.000 | 2.154 | 11.327 | 100006.344 | 100575.825 | 101369.391 |
| 9 | 0.000 | 2.558 | 10.426 | 100955.188 | 101275.212 | 101877.750 |
| ***10*** | 0.000 | 2.119 | 51.805 | 99217.750 | 101196.749 | 102382.430 |

Table B.17 Clusters information

| Cluster | Min speed | Avg speed | Max speed | Min direction | Avg direction | Max direction |
|---|---|---|---|---|---|---|
| 1 | 0.331 | 5.218 | 13.965 | 0.223 | 33.642 | 359.858 |
| 2 | 0.139 | 4.495 | 12.987 | 47.020 | 247.802 | 348.742 |
| 3 | 0.351 | 3.971 | 13.846 | 0.049 | 14.209 | 359.956 |
| 4 | 0.548 | 6.008 | 15.556 | 23.406 | 106.009 | 354.869 |
| 5 | 0.433 | 6.379 | 14.102 | 1.521 | 59.514 | 351.089 |
| 6 | 0.209 | 2.832 | 7.783 | 0.867 | 303.706 | 359.691 |
| 7 | 0.345 | 5.139 | 17.646 | 47.606 | 205.730 | 296.768 |
| 8 | 0.259 | 2.820 | 5.459 | 75.180 | 252.410 | 283.351 |
| 9 | 0.493 | 3.035 | 15.499 | 18.184 | 114.500 | 216.184 |
| ***10*** | 0.145 | 2.408 | 15.039 | 0.140 | 108.350 | 359.755 |

Table B.18 Ranges of TEMPERATURE (in °C):

| Cluster | [6.687, 9.902) | [9.902, 13.117) | [13.117, 16.331) | [16.331, 19.546) | [19.546, 22.761] |
|---|---|---|---|---|---|
| 1 | 0.00% | 31.17% | 48.55% | 15.40% | 4.88% |
| 2 | 0.71% | 26.63% | 59.23% | 8.24% | 5.18% |
| 3 | 0.07% | 42.94% | 42.94% | 10.59% | 3.46% |
| 4 | 11.09% | 31.32% | 45.09% | 12.13% | 0.37% |
| 5 | 0.23% | 31.17% | 55.55% | 12.19% | 0.86% |
| 6 | 0.00% | 28.87% | 45.63% | 21.53% | 3.97% |
| 7 | 12.50% | 61.11% | 23.61% | 2.43% | 0.35% |
| 8 | 0.00% | 52.08% | 37.50% | 10.42% | 0.00% |
| 9 | 2.08% | 89.58% | 8.33% | 0.00% | 0.00% |
| ***10*** | 2.86% | 50.36% | 36.96% | 8.04% | 1.79% |

Table B.19 Ranges of HUMIDITY (in %):

| Cluster | [0.000, 20.001) | [20.001, 40.002) | [40.002, 60.004) | [60.004, 80.005) | [80.005, 100.006] |
|---|---|---|---|---|---|
| 1 | 7.70% | 19.12% | 25.79% | 21.77% | 25.62% |
| 2 | 3.98% | 22.09% | 16.90% | 22.80% | 34.23% |
| 3 | 9.41% | 19.78% | 19.41% | 12.72% | 38.68% |
| 4 | 24.40% | 36.46% | 13.84% | 16.52% | 8.78% |
| 5 | 23.20% | 32.81% | 18.59% | 14.84% | 10.55% |
| 6 | 6.35% | 13.29% | 20.73% | 28.27% | 31.35% |
| 7 | 2.08% | 10.42% | 5.90% | 37.50% | 44.10% |
| 8 | 0.00% | 0.00% | 39.58% | 27.08% | 33.33% |
| 9 | 0.00% | 56.25% | 10.42% | 0.00% | 33.33% |
| ***10*** | 0.00% | 10.36% | 17.68% | 44.29% | 27.68% |

Table B.20 Ranges of PRECIPITATION (in Kg/mˆ2):

| Cluster | [-0.000, 10.834) | [10.834, 21.668) | [21.668, 32.502) | [32.502, 43.336) | [43.336, 54.170] |
|---|---|---|---|---|---|
| 1 | 98.59% | 1.41% | 0.00% | 0.00% | 0.00% |
| 2 | 98.86% | 1.14% | 0.00% | 0.00% | 0.00% |
| 3 | 92.57% | 7.43% | 0.00% | 0.00% | 0.00% |
| 4 | 92.19% | 2.75% | 2.23% | 2.23% | 0.60% |
| 5 | 98.28% | 1.72% | 0.00% | 0.00% | 0.00% |
| 6 | 95.24% | 4.76% | 0.00% | 0.00% | 0.00% |
| 7 | 94.44% | 0.00% | 0.69% | 1.74% | 3.12% |
| 8 | 97.92% | 2.08% | 0.00% | 0.00% | 0.00% |
| 9 | 100.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| ***10*** | 96.96% | 0.18% | 0.36% | 1.25% | 1.25% |

Table B.21 Ranges of PRESSURE (in Pa):

| Cluster | [991k, 998k) | [998k, 1005k) | [1005k, 1012k) | [1012k, 1019k) | [1019k, 102625k] |
|---|---|---|---|---|---|
| 1 | 5.96% | 18.58% | 38.00% | 25.54% | 11.92% |
| 2 | 1.14% | 4.55% | 27.56% | 40.55% | 26.21% |
| 3 | 9.41% | 10.59% | 32.94% | 30.07% | 16.99% |
| 4 | 19.94% | 41.96% | 27.38% | 4.76% | 5.95% |
| 5 | 16.33% | 48.75% | 26.17% | 6.25% | 2.50% |
| 6 | 11.11% | 6.35% | 37.10% | 29.76% | 15.67% |
| 7 | 11.11% | 0.00% | 38.89% | 38.89% | 11.11% |
| 8 | 0.00% | 66.67% | 0.00% | 33.33% | 0.00% |
| 9 | 0.00% | 0.00% | 66.67% | 33.33% | 0.00% |
| ***10*** | 14.29% | 10.36% | 18.21% | 33.75% | 23.39% |

Table B.22 Wind speed ranges (in m/s)

| Cluster | [0, 2) | [2, 4) | [4, 6) | [6, 8) | [8, 10) | [10, 12) | [12, 14) | [14, 16) | [16, 17.65] |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 4.68% | 26.24% | 35.10% | 24.34% | 7.49% | 1.70% | 0.46% | 0.00% | 0.00% |
| 2 | 15.84% | 42.05% | 15.13% | 16.76% | 2.13% | 7.03% | 1.07% | 0.00% | 0.00% |
| 3 | 11.25% | 48.38% | 27.79% | 8.09% | 3.46% | 0.44% | 0.59% | 0.00% | 0.00% |
| 4 | 5.73% | 23.21% | 25.67% | 22.92% | 11.46% | 6.77% | 2.90% | 1.34% | 0.00% |
| 5 | 4.84% | 18.98% | 26.09% | 19.61% | 17.34% | 11.33% | 1.72% | 0.08% | 0.00% |
| 6 | 27.48% | 56.05% | 11.61% | 4.86% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 7 | 13.19% | 30.56% | 31.25% | 13.19% | 1.39% | 2.08% | 2.08% | 3.12% | 3.12% |
| 8 | 33.33% | 39.58% | 27.08% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 9 | 52.08% | 22.92% | 10.42% | 8.33% | 2.08% | 2.08% | 0.00% | 2.08% | 0.00% |
| ***10*** | 58.57% | 28.21% | 6.43% | 2.14% | 1.25% | 1.79% | 1.25% | 0.36% | 0.00% |

Table B.23 Wind direction ranges

| Cluster | [, 45°) | [45°, 90°) | [90°, 135°) | [135°, 180°) | [180°, 225°) | [225°, 270°) | [270°, 315°) | [315°, 360°] |
|---|---|---|---|---|---|---|---|---|
| 1 | 95.32% | 2.87% | 0.42% | 0.00% | 0.00% | 0.00% | 0.00% | 1.39% |
| 2 | 32.39% | 64.02% | 3.41% | 0.00% | 0.00% | 0.00% | 0.00% | 0.19% |
| 3 | 0.00% | 0.00% | 0.00% | 0.28% | 5.50% | 86.38% | 7.37% | 0.47% |
| 4 | 50.20% | 0.60% | 0.10% | 0.00% | 0.00% | 0.00% | 0.60% | 48.49% |
| 5 | 1.71% | 41.94% | 55.75% | 0.40% | 0.00% | 0.00% | 0.00% | 0.20% |
| 6 | 0.45% | 0.15% | 0.00% | 0.00% | 0.00% | 0.30% | 45.68% | 53.42% |
| 7 | 0.00% | 2.30% | 95.07% | 2.63% | 0.00% | 0.00% | 0.00% | 0.00% |
| 8 | 0.22% | 0.43% | 0.00% | 0.22% | 0.00% | 26.08% | 70.91% | 2.16% |
| 9 | 0.27% | 0.00% | 0.00% | 0.00% | 8.15% | 32.61% | 51.36% | 7.61% |
| 10 | 1.17% | 0.78% | 16.02% | 78.52% | 3.52% | 0.00% | 0.00% | 0.00% |
| 11 | 0.00% | 0.42% | 0.42% | 10.00% | 83.75% | 4.58% | 0.83% | 0.00% |
| 12 | 31.25% | 10.42% | 2.08% | 0.00% | 4.17% | 6.25% | 14.58% | 31.25% |
| 13 | 0.00% | 4.17% | 8.33% | 16.67% | 16.67% | 37.50% | 16.67% | 0.00% |
| 14 | 20.83% | 20.83% | 29.17% | 20.83% | 8.33% | 0.00% | 0.00% | 0.00% |
| ***15*** | 20.19% | 10.58% | 6.25% | 7.21% | 13.46% | 7.21% | 13.46% | 21.63% |

Table B.24 Wind direction ranges

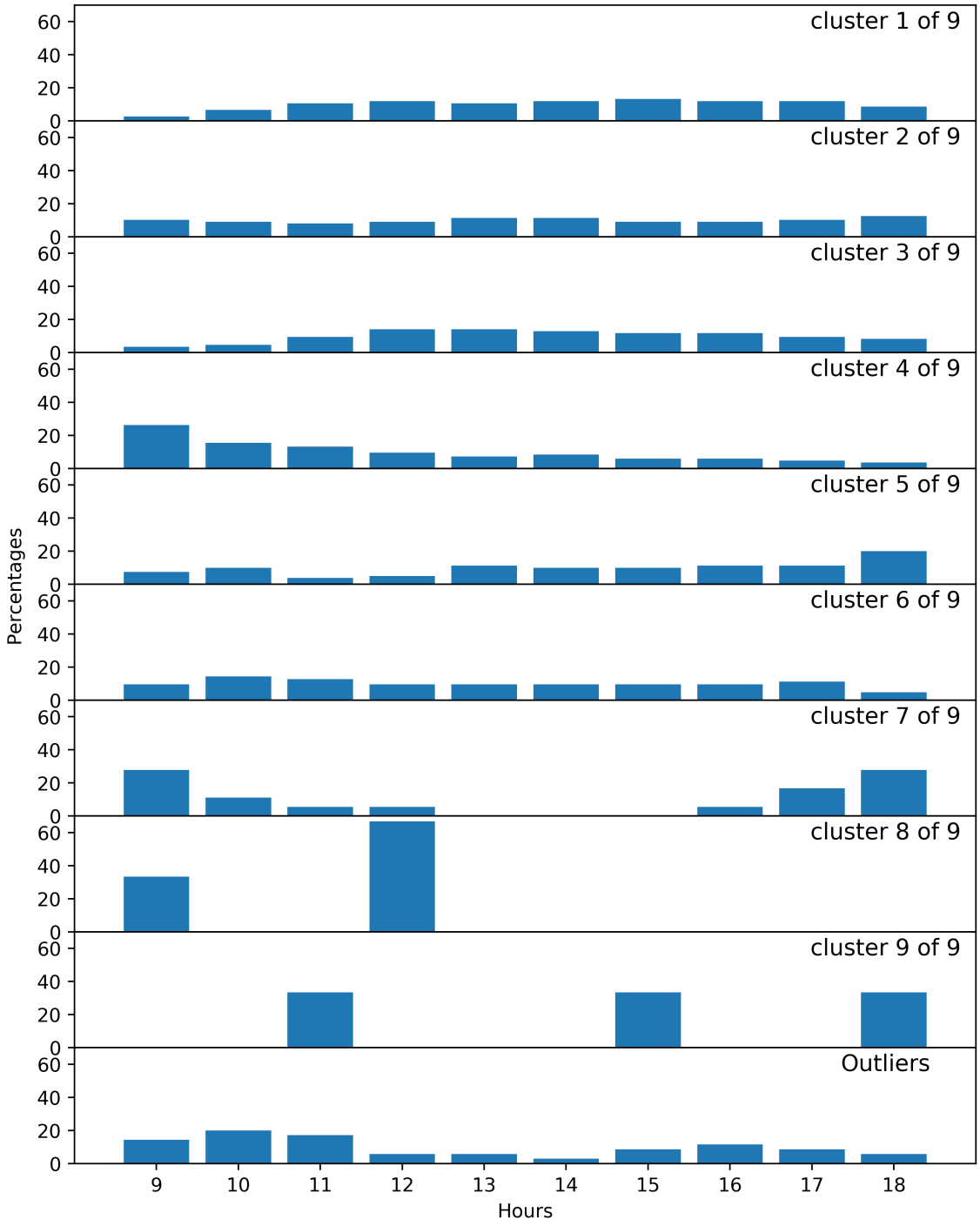
| Cluster | [0-22.5) | [22.5-45) | [45-67.5) | [67.5-90) | [90-112.5) | [112.5-135) | [135-157.5) | [157.5-180) | [180-202.5) | [202.5-225) | [225-247.5) | [247.5-270) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 25.79% | 62.46% | 9.23% | 0.95% | 0.75% | 0.04% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 2 | 0.00% | 0.00% | 0.07% | 0.00% | 0.00% | 0.00% | 0.07% | 0.14% | 0.64% | 5.04% | 42.90% | 34.52% |
| 3 | 54.04% | 5.66% | 0.44% | 0.15% | 0.07% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 4 | 0.00% | 0.60% | 1.79% | 13.24% | 50.45% | 29.99% | 3.65% | 0.22% | 0.00% | 0.00% | 0.00% | 0.00% |
| 5 | 0.31% | 9.77% | 63.28% | 18.75% | 7.58% | 0.23% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| 6 | 0.30% | 0.10% | 0.10% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.10% | 0.40% | 3.08% |
| 7 | 0.00% | 0.00% | 0.35% | 0.00% | 0.00% | 0.69% | 5.56% | 16.32% | 40.62% | 31.94% | 3.12% | 0.69% |
| 8 | 0.00% | 0.00% | 0.00% | 4.17% | 2.08% | 6.25% | 14.58% | 2.08% | 0.00% | 16.67% | 14.58% | 22.92% |
| 9 | 2.08% | 18.75% | 8.33% | 12.50% | 10.42% | 18.75% | 16.67% | 4.17% | 6.25% | 2.08% | 0.00% | 0.00% |
| ***10*** | 5.54% | 6.43% | 3.39% | 3.04% | 2.14% | 6.43% | 16.79% | 9.29% | 1.43% | 5.54% | 4.82% | 6.25% |

Table B.25 Wind direction wider ranges

| Cluster | [270-292.5) | [292.5-315) | [315-337.5) | [337.5-360] |
|---|---|---|---|---|
| 1 | 0.00% | 0.00% | 0.12% | 0.66% |
| 2 | 13.49% | 2.34% | 0.57% | 0.21% |
| 3 | 0.07% | 0.51% | 7.21% | 31.84% |
| 4 | 0.00% | 0.00% | 0.00% | 0.07% |
| 5 | 0.00% | 0.00% | 0.00% | 0.08% |
| 6 | 26.98% | 34.92% | 29.17% | 4.86% |
| 7 | 0.35% | 0.35% | 0.00% | 0.00% |
| 8 | 16.67% | 0.00% | 0.00% | 0.00% |
| 9 | 0.00% | 0.00% | 0.00% | 0.00% |
| ***10*** | 7.86% | 8.21% | 7.14% | 5.71% |

Table B.26 Transition matrix

| Cluster | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 79.71 | 0.00 | 50.00 | 3.57 | 46.43 | 0.00 | 0.00 | 0.00 | 0.00 |
| 2 | 0.00 | 84.93 | 0.00 | 0.00 | 0.00 | 54.55 | 36.36 | 0.00 | 9.09 |
| 3 | 65.22 | 0.00 | 70.13 | 0.00 | 0.00 | 30.43 | 0.00 | 0.00 | 4.35 |
| 4 | 31.82 | 0.00 | 0.00 | 71.05 | 68.18 | 0.00 | 0.00 | 0.00 | 0.00 |
| 5 | 73.68 | 0.00 | 0.00 | 26.32 | 70.31 | 0.00 | 0.00 | 0.00 | 0.00 |
| 6 | 0.00 | 52.94 | 47.06 | 0.00 | 0.00 | 70.69 | 0.00 | 0.00 | 0.00 |
| 7 | 0.00 | 100.00 | 0.00 | 0.00 | 0.00 | 0.00 | 66.67 | 0.00 | 0.00 |
| 8 | 0.00 | 50.00 | 0.00 | 0.00 | 0.00 | 50.00 | 0.00 | 0.00 | 0.00 |
| 9 | 0.00 | 0.00 | 100.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |


cluster 1 of 9
cluster 2 of 9
cluster 3 of 9
cluster 4 of 9
cluster 5 of 9
cluster 6 of 9
cluster 7 of 9
cluster 8 of 9
cluster 9 of 9
Outliers
Percentages
0
20
40
60
9
10
11
12
13
14
15
16
17
18
Hours

### B.4.2 Comparison with automatic clustering

Table B.27 Clusters matching

| Cluster automatic | Cluster manual k = 10 | Common elements |
|---|---|---|
| 1 (87) | 1 (151) | 87 |
| 4 (55) | 4 (84) | 54 |
| 2 (68) | 3 (85) | 39 |
| 7 (35) | 5 (80) | 35 |
| 8 (34) | 2 (88) | 34 |
| 10 (28) | 6 (63) | 28 |
| 14 (19) | 7 (18) | 17 |
| 15 (16) | 8 (3) | 2 |
| 3 (60) | 9 (3) | 0 |

Maximum Matching Measure = 0.485

### B.4.3 Comparison with manual clustering k = 17

Table B.28 Clusters matching

| Cluster manual k = 10 | Cluster manual k = 17 | Common elements |
|---|---|---|
| 1 (151) | 1 (135) | 115 |
| 2 (88) | 3 (67) | 67 |
| 3 (85) | 4 (62) | 62 |
| 5 (80) | 2 (99) | 62 |
| 4 (84) | 5 (62) | 43 |
| 6 (63) | 6 (42) | 39 |
| 7 (18) | 11 (15) | 15 |
| 8 (3) | 13 (3) | 3 |
| 9 (3) | 14 (3) | 3 |

Maximum Matching Measure = 0.670